PATENT
Attorney Docket No.: SONY-29300

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	Group Art Unit: 2434
Edward Eytchison et al.	)	Examiner: Jung, David Yiuk
Serial No.: 10/763,654	)	
Filed: January 22, 2004	)	**APPEAL BRIEF**
For: **METHOD AND APPARATUS FOR DETERMINING AN IDENTITY OF A USER**	)	162 North Wolfe Road Sunnyvale, California 94086 (408) 530-9700
	)	Customer No. 28960

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In furtherance of the Applicants' Notice of Appeal filed on January 16, 2009, this Appeal Brief is submitted. This Appeal Brief is submitted in support of the Applicants' Notice of Appeal, and further pursuant to the rejection mailed on October 16, 2008, in which Claims 1-34 were rejected. The Applicants submit this Appeal Brief to the Board of Patent Appeals and Interferences in compliance with the requirements of 37 C.F.R. § 41.37, as stated in *Rules of Practice Before the Board of Patent Appeals and Interferences (Final Rule)*, 69 Fed. Reg. 49959 (August 12, 2004). The Applicants contend that the rejections of Claims 1-34 in this proceeding are in error, were previously overcome and are overcome again by this appeal.

I. REAL PARTIES IN INTEREST

As the assignee of the entire right, title, and interest in the above-captioned patent application, the real parties in interest in this appeal, is:

Sony Corporation, a Japanese corporation
6-7-35 Kitashinagawa, Shinagawa
Tokyo, 141
Japan

Sony Electronics Inc., a corporation of the State of Delaware
1 Sony Drive
Park Ridge, NJ 07656-8003

per the assignment document filed on January 22, 2004.

II. RELATED APPEALS AND INTERFERENCES

The Applicants are not aware of any other appeals or interferences related to the present application.

III. STATUS OF THE CLAIMS

Claims 1-34 are involved in the appeal. Claims 1-21 and 25-34 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Darrell et al., "Integrated person tracking using stereo, color, and pattern detection" ("Darrell," a copy of which is attached as Exhibit A) in view of Davis et al., "Context Tailor: Towards a Programming Model for Context-Aware Computing," ("Davis," a copy of which is attached as Exhibit B). Claims 22-24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Darrell and Davis in view of Seno et al., "Network authentication system with multi-biometric," ("Seno" a copy of which is attached as Exhibit C).

IV. STATUS OF THE AMENDMENTS FILED AFTER FINAL REJECTION

No amendments to the claims have been filed after the Office Action mailed on October 16, 2008.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The invention disclosed in the present application number 10/763,654 is directed to methods and apparatuses for determining an identity of a user. The identity is determined by detecting a current user's electronic device activity pattern; comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular user; and then using the comparing to identify the current user as being the particular user.

The elements of Claim 1, directed to one embodiment, are described in the Specification at page 10, line 1 to page 13, line 14, page 15, line 16 to page 19, line 10, page 19, line 11 to page 21, line 2, and accompanying Figures 3, 5 and 6. The method of identifying a user comprises detecting a user's electronic device (110) activity pattern (510), comparing the detected activity pattern against a plurality of user action identification profiles stored in a memory device (520), wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user and using the comparing to identify the current user as being one of the particular users (530, 540, 550).

The elements of Claim 10, directed to one embodiment, are described in the Specification at page 7, line 16 to page 8, line 16, page 8, line 17 to page 9, line 22, page 10, line 1 to page 13, line 14, and accompanying Figures 1, 2 and 3. The system comprises means for detecting (320) a user's electronic device (110) activity pattern, means for comparing (330) the detected activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user and means for using (330) the comparing to identify the current user as being one of the particular users.

Means for detecting a referred to in the specification as a detection module (320) is shown in Figure 3. Detection module (320) is configured to monitor the user's activity on electronic device (110). [Present Specification, page 10, lines 9-15]

Means for comparing referred to in the specification as comparator module (330) is shown in Figure 3. Comparator module (330) is configured to compare activity being performed by an electronic device 110 user against at least one user action identification profile stored by database module (340). [Present Specification, page 12, lines 5-7]

Means for using referred to in the specification as comparator module (330) is shown in Figure 3. The comparator module (330) uses a predetermined threshold score to determine the

match quality between a particular user action identification profile and the activity(ies) current being performed by a user. [Present Specification, page 12, lines 16-18]

The elements of Claim 11, directed to one embodiment, are described in the Specification at page 10, line 1 to page 13, line 14, page 15, line 16 to page 19, line 10, page 19, line 11 to page 21, line 2, and accompanying Figures 3, 5 and 6. The method comprises comparing a user's activity pattern against a plurality of user action identification profiles stored in a memory device (520), wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user and wherein the current user's activity includes an input selection and using the comparing to identify the current user as being one of the particular users (530, 540, 550).

The elements of Claim 21, directed to one embodiment, are described in the Specification at page 19, line 11 to page 21, line 2, and accompanying Figure 6.. The method comprises determining a particular user's identity (610), detecting the particular user's activity pattern (620) and storing the particular user's activity pattern within a user action identification profile in a memory device (630), wherein the user action identification profile is configured to be compared with an unknown user's activity pattern against at least one activity performed by the particular unique user (640).

The elements of Claim 27, directed to one embodiment, are described in the Specification at page 7, line 16 to page 8, line 16, page 8, line 17 to page 9, line 22, page 10, line 1 to page 13, line 14, and accompanying Figures 1, 2 and 3. The identification system comprises a detection module (320) configured for detecting a user's activity pattern (510) and a comparator module (330) configured for comparing the user's activity pattern to a user action identification profile stored within a memory device (520), wherein the comparator module (330) is configured to determine a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile comprising at least one activity performed by the user (530, 540, 550).

The elements of Claim 32, directed to one embodiment, are described in the Specification at page 10, line 1 to page 13, line 14, page 15, line 16 to page 19, line 10, page 19, line 11 to page 21, line 2, and accompanying Figures 3, 5 and 6. The method comprises detecting a user's electronic device (110) activity pattern (520, 620), storing the user's activity pattern in a memory storage device within a user action identification profile comprising at least one activity by the user (630), comparing the detected activity pattern against a plurality of user action identification profiles (520), wherein each user action identification profile is associated with a particular

unique user, using the comparing to identify the current user as being one of the particular users (530-550) and continuing to update the user's stored activity pattern after identifying the user (610-640).

The elements of Claim 33, directed to one embodiment, are described in the Specification at page 10, line 1 to page 13, line 14, page 15, line 16 to page 19, line 10, page 19, line 11 to page 21, line 2, and accompanying Figures 3, 5 and 6. The identification system comprises a detection module (320) to detect a user's activity pattern (510), a storage module (340) to store the user's activity pattern within a user action identification profile comprising at least one activity by the user (630) and a comparator module (330) to compare the user's activity pattern to the user action identification profile (520), wherein the comparator module (330) determines a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile (530, 540, 550).

VI. GROUNDS OF REJECTION AND OTHER MATTERS TO BE REVIEWED ON APPEAL

The following issues are presented in this Appeal Brief for review by the Board of Patent Appeals and Interferences:

1. Whether Claims 1-21 and 25-34 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Davis and Darrell.
2. Whether Claims 22-24 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Darrell and Davis in view of Seno.

VII. ARGUMENT

Grounds for Rejection

Within the Office Action, Claims 1-21 and 25-34 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Davis and Darrell.

Outline of Arguments

In the discussion that follows, the Applicants discuss the teachings of Davis, the teachings of Darrell and the teachings of the combination of Davis and Darrell. As will be discussed in detail below, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each

user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. As further discussed below, the combination of Davis and Darrell is improper.

1. Davis does not teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. Davis also does not teach identifying the current user as being one of the particular users.

Davis teaches a programming model for customized context-aware applications. Specifically, Davis teaches that applications customize their execution to expected needs of a user based on patterns of repetitive context. [Davis, section 1] As recognized within the Office Action of October 16, 2008, however, Davis does not teach identifying the current user as being one of the particular users. [Davis, section 1, para. 1] This is because the purpose of Davis is to design applications that customize themselves based on patterns of use regardless of the user. In other words, the programs track how they are used and adjust accordingly even if there is a different user each time. As a result, Davis does not care who the user is or even the number of users, so there is certainly no need for Davis to distinguish between the current user and any particular user. Furthermore, Davis does not teach comparing the detected activity pattern against a plurality of user action identification profiles. Davis merely teaches classifying an event as recurring or rare, if the event falls within a pattern, and the length of the event. [Davis, section 3, para. 1] In fact, because at no point does Davis track the identification of any one user, it is impossible for Davis to store user action identification profiles, much less compare the detected activity pattern against them. As a result, Davis does not teach comparing the detected activity pattern against a plurality of user action identification profiles.

2. Darrell does not teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user.

Darrell teaches a visual person tracking system. Darrell teaches tracking persons in a video scene using three visual processing modules, each for conducting depth estimation, detecting color segmentation and discriminating head regions from hands and other body parts. [Darrell, section 2, para. 2] Darrell teaches that stereo cameras are used to observe the attributes of a person and the observed attributes are compared with stored statistics of previous tracked users. [Darrell, section 5.2, para. 1] In one embodiment, Darrell teaches that the stored attributes include a face pattern, height, and color observations of a user. [Darrell, section 5.2, para. 2] However, like Davis, Darrell does not teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user. Darrell merely teaches comparing invariable attributes such as face patterns, height and color observations of previous tracked users, not actions or other dynamic data.

3. The combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. Even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. The combination of Davis and Darrell is also improper because they are nonanalogous art. Further, even if their combination is proper, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user.

This is a classic case of impermissibly using hindsight to make a rejection based on obviousness. The Court of Appeals for the Federal Circuit has stated that "it is impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings

of the prior art so that the claimed invention is rendered obvious." In Re Fritch, 972 F.2d, 1260, 1266, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992). As recognized within the Office Action of October 16, 2008, Davis does not teach to identify the current user as being one of the particular users. Within the Office Action of October 16, 2008, it is stated that

> [i]t would have been obvious for one of ordinary skill in the art at the time of the to modify the claimed invention to combine the teachings of Darrell and Davis for the [motivation of better tracking persons]. [Office Action, page 5]

However, it is only with the benefit of the present claims, as a "template" that there is any motivation to combine the identification of a particular user of Darrell with the monitoring of the use of applications of Davis. No such motivation can be found in the teachings of either of the references. To conclude that the combination of Davis and Darrell is obvious, based on the teachings of these references, is to use hindsight based on the teachings of the presently claimed invention and to read much more into Davis and Darrell than their actual teachings. This is simply not permissible based on the directive from the Court of Appeals for the Federal Circuit.

It is well settled that to establish a *prima facie* case of obviousness, three basic criteria must be met:

1) there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings;
2) there must be a reasonable expectation of success; and
3) the prior art reference, or references, must teach or suggest all the claim limitations. MPEP § 2143.

The burden of establishing a *prima facie* case of obviousness based on the teachings of Davis and Darrell has not been met within the Office Action of October 16, 2008.

There is no hint, teaching or suggestion within either Davis or Darrell that justifies their combination. Within the Office Action of October 16, 2008, the improper combination's only justification is that it would allow the better tracking of persons. [Office Action, page 5] The Applicants respectfully disagree. No citation from either Davis or Darrell is provided within the Office Action of October 16, 2008 that comes close to supporting this improper conclusion. More is required to justify the combination of two references. As described above, Davis is not concerned with identifying or tracking the user because the customization of the applications is

performed based on the actions of any and all users of the application. Therefore, incorporating the better tracking of persons provided by Darrell would not provide any benefit to Davis. Thus, there is simply no hint, teaching or suggestion within either of these references that warrants or justifies their combination. Accordingly, for this reason alone the combination of Davis and Darrell is improper.

However, Davis and Darrell also cannot be properly combined because the combination would change their principle operation. Specifically, the MPEP states "[i]f the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious." In re Ratti, 270 F.2d 810, 123 (CCPA 1959); MPEP §2143.01. The principle operation of Davis is to input patterns of repetitive context (i.e., manners of interaction with an application) in order for that application to self customize based on that context regardless of the user or combination of users who did the interacting. [See Davis, sections 1 and 3] To that end, the interaction data collected in Davis is commingled to form a single large data pool used to create the ideal application for the average user. [Id.] In other words, the applications of Davis are designed to self customize based on interaction with users regardless of any one particular user's interaction. In contrast, the principle operation of Darrell involves a passive tracking device, wherein data specific to each "user" is collected and kept separate in multiple smaller data pools. [See Darrell, section 2] Further, in the principle operation of Darrell these specific "users" do not interact with the device, but instead the device monitors how the "users" interact around the device in order to identify them. In other words, the principle operation of Darrell involves specifically tracking each user without the user interacting with the device. Thus, in order to modify Davis to include Darrell or Darrell to include Davis, their principle methods of operation would need to be changed. Either, Davis would need to be altered so that it both tracked each user and monitored user action around the application, but not with the application, or Darrell would need to be altered so that it both input data regardless of the user and provided a way for the user's to interact with the device. It is clear that either modification would completely change the principle operation of each reference. A passive tracking device like Darrell would never be able to interact with the persons (e.g. criminals) it was trying to track and an application self customization process like Davis would never be able to collect the passive data of the user's actions around the application nor would such actions be applicable to the desired customization.

Indeed, as described above, the passive observation of Darrell would be of no value to Davis because although it might be able to recognize a particular user, its self-customization is independent of the users. Thus, the same output would be produced regardless of what user is identified as using the application at the current time. In other words, the system of Davis would derive no value from the identification capabilities of Darrell, and thus has no motivation to incorporate it, because it would have no use for the information that Darrell provides. Indeed, the only way for Davis to benefit from Darrell's user identification capabilities would be to completely change its self-customization technique to center around each user instead of the frequency of occurrence regardless of the user.

Such a change would dramatically alter the goals or purpose of Davis. By self customizing regardless of the user, Davis is able to input a large amount of data and therefore become the "ideal" application for the "average" user even if it is a first-time user. However, in order for Davis to change such that the identity information of Darrell would be beneficial, Davis would have to separate the customization data into multiple smaller data pools for each user, thereby significantly lowering the amount of data for, and correspondingly the quality of, the customization. Additionally, Davis would become completely "uncustomized" for any first-time users. These are two, distinct and incompatible purposes. With its current purpose of objective customization, Davis would be "ideal" for applications run on a public library computer because each first-time user would derive a benefit from the information collected from all the previous users. In order to utilize the information provided by Darrell, the purpose of Davis would have to be subjective customization, which would then render Davis completely ineffective for a public library computer where there are numerous first-time users. Accordingly, the combination of Darrell and Davis is also improper because it would alter their principle modes of operation.

Moreover, the combination of Davis and Darrell is improper because to rely on a reference under 35 U.S.C. §103, it must be analogous prior art according to MPEP 2141.01(a). Specifically, "the reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." In re Oetiker, 977 F.2d 1443, 1446 (Fed. Cir. 1992). Here, if the field of the applicant's endeavor is construed to be identifying particular users, then Davis is nonanalogous art because it does not relate to that field nor does it try and solve that problem. In fact, as described above, Davis is designed to self customize the application regardless of any particular user. Additionally, if the field of the applicant's endeavor is construed to be inputting interaction data, then Darrell is nonanalogous art because it does not interact with any users nor does it try to solve such a

problem. In fact, as described above, it is highly unlikely that a monitoring device like Darrell would be able to interact with its "users." The passive tracking system of Darrell is not the same art as the interactive application customization device of Davis, nor do they attempt to solve the same problems. Accordingly, either Davis or Darrell will be nonanalogous art and as a result, Davis and Darrell are not able to be combined to form a proper §103 rejection.

In contrast to Davis, Darrell and their combination, the presently claimed invention is directed to methods and apparatuses for determining an identity of a user. The identity is determined by detecting a current user's electronic device activity pattern; comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular user; and then using the comparing to identify the current user as being the particular user. As described above, even if considered proper, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. Also as described above, there is no motivation to combine the teachings of Davis with the teachings of Darrell. Furthermore, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover, Davis and Darrell cannot be properly combined because they are nonanalogous art. Accordingly, the combination of Davis and Darrell is improper.

4. The claims distinguish over Davis, Darrell and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 1-9

The independent Claim 1 is directed to a method of identifying a user comprising detecting a user's electronic device activity pattern, comparing the detected activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user, and using the comparing to identify the current user as being one of the particular users. Within Claim 1 it is recited that the detected activity pattern is

compared against a plurality of user action identification profiles. It is further recited in Claim 1 that each user action identification profile is associated with a particular unique user, and the current user is identified as being one of the particular users. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 1 is allowable over the teachings of Darrell, Davis and their combination.

Claims 2-9 are all dependent upon the independent Claim 1. As discussed above, the independent Claim 1 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the Claims 2-9 are all also allowable as being dependent upon an allowable base claim.

b. Claim 10

The independent Claim 10 is directed to a system comprising means for detecting a user's electronic device activity pattern, means for comparing the detected activity pattern against a plurality of user action identification profiles stored within a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user, and means for using the comparing to identify the current user as being one of the particular users. Within Claim 10 it is recited that the means for comparing is used to compare the detected activity pattern against a plurality of user action identification profiles stored in a memory device. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell

nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 10 is allowable over the teachings of Darrell, Davis and their combination.

c. Claims 11-20

The independent Claim 11 is directed to a method comprising comparing a user's activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user and wherein the current user's activity includes an input selection and using the comparing to identify the current user as being one of the particular users. Within Claim 11 it is recited that the user's activity pattern is compared against a plurality of user action identification profiles and that the user's activity includes an input selection. It is further recited in Claim 11 that each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user, and the current user is identified as being one of the particular users. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 11 is allowable over the teachings of Darrell, Davis and their combination.

Claims 12-20 are all dependent upon the independent Claim 11. As discussed above, the independent Claim 11 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the Claims 12-20 are all also allowable as being dependent upon an allowable base claim.

d. Claims 21, 25 and 26

The independent Claim 21 is directed to a method comprising determining a particular user's identity, detecting the particular user's activity pattern, and storing the particular user's activity pattern within a user action identification profile in a memory device, wherein the user action identification profile is configured to be compared with an unknown user's activity pattern against at least one activity performed by the particular unique user. Within Claim 21 it is recited that an unknown user's activity pattern is compared against a stored user action identification profile. It is further recited in Claim 21 that the stored user action identification profile includes a particular detected user's activity pattern. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 21 is allowable over the teachings of Darrell, Davis and their combination.

Claims 25 and 26 are both dependent upon the independent Claim 21. As discussed above, the independent Claim 21 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the Claims 25 and 26 are both also allowable as being dependent upon an allowable base claim.

e. Claims 27-31

The independent Claim 27 is directed to an identification system comprising a detection module configured for detecting a user's activity pattern, and a comparator module configured for comparing the user's activity pattern to a user action identification profile stored within a memory device, wherein the comparator module is configured to determine a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile comprising at least one activity performed by the user. Within Claim 27 it is recited that a

comparator module is configured to determine a user's identity based on scoring a comparison between the user's activity pattern and a user action identification profile comprising at least one activity performed by the user. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 27 is allowable over the teachings of Darrell, Davis and their combination.

Claims 28-31 are all dependent upon the independent Claim 27. As discussed above, the independent Claim 27 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the Claims 28-31 are all also allowable as being dependent upon an allowable base claim.

f. Claim 32

The independent Claim 32 is directed to a method comprising detecting a user's electronic device activity pattern, storing the user's activity pattern in a memory storage device within a user action identification profile comprising at least one activity by the user, comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user, using the comparing to identify the current user as being one of the particular users, and continuing to update the user's stored activity pattern after identifying the user. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a

plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 32 is allowable over the teachings of Darrell, Davis and their combination.

g. Claims 33 and 34

The independent Claim 33 is directed to an identification system comprising a detection module to detect a user's activity pattern, a storage module to store the user's activity pattern within a user action identification profile comprising at least one activity by the user, and a comparator module to compare the user's activity pattern to the user action identification profile, wherein the comparator module determines a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile. As described above, the combination of Davis and Darrell is improper because there is no motivation to combine the teachings of Davis with the teachings of Darrell. As further described above, even if there were motivation to combine Davis and Darrell the combination is impermissible because it would result in a drastic change in their principle operation. Moreover as described above, Davis and Darrell cannot be properly combined because they are nonanalogous art. Further, as described above, neither Davis, Darrell nor their combination teach comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user by an activity performed by the particular unique user. For at least these reasons, the independent Claim 33 is allowable over the teachings of Darrell, Davis and their combination.

Claim 34 is dependent upon the independent Claim 33. As discussed above, the independent Claim 33 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the Claim 34 is also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Final Office Action, Claims 22-24 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Darrell and Davis in view of Seno.

Arguments

Claims 22-24 are all dependent upon the independent Claim 21. As discussed above, the independent Claim 21 is allowable over the teachings of Darrell, Davis and their combination. Accordingly, the dependent Claims 22-24 are all also allowable as being dependent upon an allowable base claim.

4. CONCLUSION

For the above reasons, it is respectfully submitted that the Claims 1-34 are allowable over the cited prior art references. Therefore, a favorable indication is respectfully requested.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: March 13, 2009

By: /Jonathan O. Owens/
Jonathan O. Owens
Reg. No.: 37,902
Attorney for Applicant

VIII. CLAIMS APPENDIX

This appendix includes a list of the claims under appeal.

1. A method of identifying a user comprising:
detecting a user's electronic device activity pattern;
comparing the detected activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user; and
using the comparing to identify the current user as being one of the particular users.

2. The method according to claim 1, wherein comparing the detected activity pattern comprises scoring a comparison between the detected activity pattern and a user action identification profile.

3. The method according to claim 1, wherein comparing the user activity pattern comprises scoring a comparison between the detected activity pattern and a user action identification profile, and wherein using the comparing comprises comparing the comparison score against a predetermined threshold score.

4. The method according to claim 1, wherein comparing the detected activity pattern comprises determining a number of matches between the detected activity pattern and the user action identification profiles, and wherein using the comparing comprises comparing the determined number of matches against a predetermined number of matches.

5. The method according to claim 1, further comprising the act of detecting additional activity of the current user if the act of using the comparing does not identify the current user.

6. The method according to claim 1, wherein the current user's electronic device activity includes selection of content.

7. The method according to claim 1, wherein the current user's electronic device activity includes selection of an application.

8. The method according to claim 1, wherein the activity of the user includes selection of a category.

9. The method according to claim 1, wherein the detected activity pattern includes a length of time between the current user's inputs on the electronic device.

10. A system comprising:
means for detecting a user's electronic device activity pattern;
means for comparing the detected activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user; and
means for using the comparing to identify the current user as being one of the particular users.

11. A method comprising:
comparing a user's activity pattern against a plurality of user action identification profiles stored in a memory device, wherein each user action identification profile is associated with a particular unique user by at least one activity performed by the particular unique user and wherein the current user's activity includes an input selection; and
using the comparing to identify the current user as being one of the particular users.

12. The method according to claim 11, wherein comparing the current user's activity pattern comprises scoring a comparison between the detected activity pattern and a user action identification profile.

13. The method according to claim 11, wherein comparing the current user's activity pattern comprises scoring a comparison between the current user's activity pattern and a user action identification profile, and wherein using the comparing comprises comparing the comparison score against a predetermined threshold score.

14. The method according to claim 11, wherein comparing the current user's activity pattern comprises determining a number of matches between the current user's activity pattern and the user action identification profiles, and wherein using the comparing comprises comparing the determined number of matches against a predetermined number of matches.

15. The method according to claim 11, further comprising the act of detecting additional activity of the current user if the act of using the comparing does not identify the current user.

16. The method according to claim 11, wherein the input selection includes selection of content.

17. The method according to claim 11, wherein the input selection includes selection of an application.

18. The method according to claim 11, wherein the input selection includes selection of a category.

19. The method according to claim 11, wherein the current user's activity pattern includes a length of time between the current user's inputs.

20. The method according to claim 11, further comprising detecting the user's activity pattern.

21. A method comprising:
determining a particular user's identity;
detecting the particular user's activity pattern; and

storing the particular user's activity pattern within a user action identification profile in a memory device, wherein the user action identification profile is configured to be compared with an unknown user's activity pattern against at least one activity performed by the particular unique user.

22. The method according to claim 21, wherein the determining the particular user's identity further comprises detecting a particular user's biometric parameter.

23. The method according to claim 22, wherein the user's biometric parameter includes one of an iris scan, a DNA sample, and a fingerprint.

24. The method according to claim 21, wherein the determining the particular user's identity further comprises detecting a particular user's password.

25. The method according to claim 21, wherein the determining the particular user's identity further comprises comparing the particular user's activity pattern with the user action identification profile.

26. The method according to claim 25, wherein confirming the identity of the particular user further comprises scoring a sufficient match between the activity of the particular user with profile data associated with a known user in response to comparing the activity.

27. An identification system comprising:
a detection module configured for detecting a user's activity pattern; and
a comparator module configured for comparing the user's activity pattern to a user action identification profile stored within a memory device, wherein the comparator module is configured to determine a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile comprising at least one activity performed by the user.

28. The system according to claim 27, further comprising a database module configured for storing the user action identification profile.

29. The system according to claim 27, wherein the user's activity pattern includes one of selection of content, selection of an application, and selection of a category.

30. The system according to claim 27, wherein the comparator module is configured to compare a score comparison against a predetermined threshold score.

31. The system according to claim 27, wherein the comparator module is configured to determine a best comparison score based on the user's activity pattern for a predetermined length of time.

32. A method comprising:
detecting a user's electronic device activity pattern;
storing the user's activity pattern in a memory storage device within a user action identification profile comprising at least one activity by the user;
comparing the detected activity pattern against a plurality of user action identification profiles, wherein each user action identification profile is associated with a particular unique user;
using the comparing to identify the current user as being one of the particular users; and
continuing to update the user's stored activity pattern after identifying the user.

33. An identification system comprising:
a detection module to detect a user's activity pattern;
a storage module to store the user's activity pattern within a user action identification profile comprising at least one activity by the user; and
a comparator module to compare the user's activity pattern to the user action identification profile, wherein the comparator module determines a user's identity based on scoring a comparison between the user's activity pattern and the user action identification profile.

34. The system according to claim 33, wherein the storage module continues to update the user's stored activity pattern after the user is identified.

IX. EVIDENCE APPENDIX

STATEMENT

Pursuant to 37 C.F.R. § 41.37(c)(1)(ix), the following is a statement setting forth where in the record the evidence of this appendix was entered by the examiner:

Evidence Description:	**Where Entered:**
Davis et al., "Context Tailor: Towards a Programming Model for Context-Aware Computing."	Office Action mailed February 22, 2007
Seno et al., "Network authentication system with multi-biometric."	Office Action mailed February 22, 2007
Darrell et al., "Integrated person tracking using stereo, color, and pattern detection."	Office Action mailed February 22, 2007
Office Action October 16, 2008	Examiner Office Action

X. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.

International Middleware Conference Workshop Proceedings – Middleware for Pervasive and Ad Hoc Computing, 16-20 June 2003, Rio de Janeiro, Brazil, 68 – 75, © 2003 ACM/IFIP/USENIX

Context Tailor: Towards a Programming Model for Context-Aware Computing

John S. Davis II, Daby M. Sow, Marion Blount and Maria R. Ebling
IBM T. J. Watson Research Center
19 Skyline Drive
Hawthorne, NY, 10532 - USA

Abstract

Many context-aware computing applications form inferences and execute corresponding actions based on context that is uniquely associated with a user. We refer to such applications as customized context-aware applications and recognize that their design poses a very challenging burden to application designers due to the degree of customization that is required. To tackle this problem, we propose to develop a programming model and framework for context-aware applications with the goal of shielding application developers from the complexity of customization. The framework applies machine learning in novel ways to infer application triggering conditions. This paper presents the key research challenges that must be overcome to reach this goal along with the directions that we are currently exploring.

1. Introduction

Pervasive, context-aware computing is the process of using pervasive data from the user's environment (i.e., context) to adapt the execution of a computation on a user's behalf. The Context Tailor project focuses on an important class of pervasive, context-aware applications that we refer to as *customized context-aware* (CCA) applications. Such applications customize their execution to the expected needs of the user based on patterns of repetitive context. Examples include:

- Weiser's waking state coffee machine [9] that brews coffee in anticipation of a user waking up.
- A UPS truck assignment application that predicts recipient availability for efficient delivery.
- A Smart HVAC system that adjusts the thermostat just-in-time by predicting a user's arrival.
- A scheduler that predicts computer idleness to schedule expensive computations conveniently.
- A content distribution system that pre-fetches/pre-transcodes web content by predicting user access based on user context.

Customized context-aware applications represent an important emerging class of new applications within pervasive computing that involve predictions of user behavior based on context about the user. These applications blend the agenda of pervasive computing with machine learning. Users benefit from these applications in that the applications facilitate calm computing [10] by enabling mundane tasks to be performed without requiring conscious interaction by a user and by preparing a service for a user's anticipated needs. An additional, but arguably more significant, benefit is that the removal of active user attention from the execution process affords the opportunity for extremely efficient application execution.

Imagine developing a CCA application such as Weiser's waking state coffee machine that prepares coffee so that it is ready at the appropriate time in the morning. The challenge faced by the designer of this application is the meaning of appropriate time. Some users want coffee immediately upon waking while others want it after taking their morning shower. For both of these cases, the developer must either determine the right context conditions for triggering the coffee maker (e.g., should restless sleep or physical absence from the bed signal that coffee will soon be desired?) or incorporate user preferences to specify the appropriate conditions. In either case, customization according to user characteristics is a considerable design burden. The problem is exacerbated when one considers that a user's characteristics change over time requiring new inferences to be specified. If either the developer or the end user is required to spend significant time customizing or tailoring an application to meet each user's needs, few such applications will reach the market and even fewer will be successful.

The Context Tailor project is striving to radically simplify the development of CCA applications by pushing application triggering functionality out of the developer's view and into middleware so that application developers are shielded from the complexity of customization. To achieve

this goal we are placing a machine learning infrastructure into the middleware.

Pervasive computing middleware is being developed by several research and commercial entities to overcome the challenges that pervasive computing environments pose [1][2][14]. Such challenges include heterogeneous data formats, temporarily disconnected networks, variable reliability, privacy and the challenge of specifying complex data compositions. In a separate vein, machine learning as an application-level facility within pervasive computing has been attempted by several research efforts such as the MavHome project [15]. Machine learning as a component of pervasive computing middleware is the contribution of this paper. Our vision is to develop middleware that offers machine learning capabilities without requiring application developers to be machine learning experts. Furthermore, we expect the machine learning engine to overcome the challenges that pervasive computing imposes. Our goal can be summarized by the slogan *write once, run for everyone.*

We see three major research challenges that must be addressed in order to achieve this ideal:

1. The definition of a programming model for customized context-aware applications.
2. The design and evaluation of a generic inference framework based upon machine-learning techniques.
3. An approach for managing privacy in a tractable manner.

In what follows, we present these challenges after a brief presentation of our context-aware platform and our targeted application space.

2. Existing Context Service

Context Tailor middleware is designed on top of an existing Context Service [3] that was developed at IBM research. The role of the Context Service is to collect, maintain and provide context information to applications about numerous *subjects*. Subjects may be users or objects (e.g., equipment or packages).

Figure 1 shows a high level description of the Context Service. It consists of a dispatcher, a configurable set of context drivers, and a collection of utility components. In addition, there are three programming interfaces: a Client API, a Context Push Interface and an internal Context Driver Interface. The dispatcher routes application requests for context to the appropriate context drivers through the Context Driver Interface. Each of these context drivers handles only one type of context information and encapsulate the details of the interaction with context sources of that type of context information. Examples of context include location, calendar activity and network connectivity. Context Drivers may either pull information from context sources or let context sources push information to them via the Context Push Interface.

With the Client API, applications may either poll the Context Service for context information or subscribe for notifications of new context data published by the appropriate context drivers. The Context Tailor project is an application of the Context Service that accesses context information through the Client API and identifies context patterns that can be used by context-aware applications.

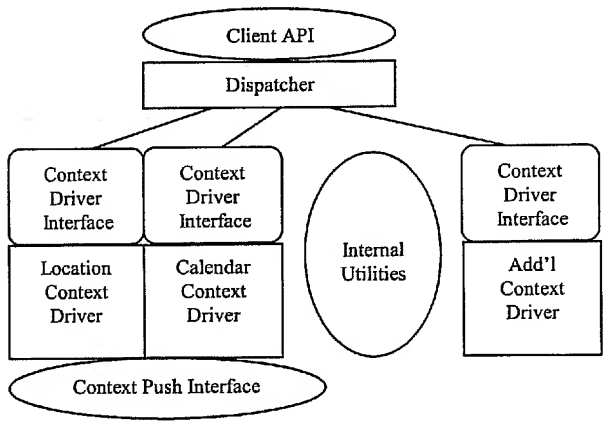

Figure 1: Context Service Architecture

3. Application Space

Figure 2 displays a classification of customized context-aware applications based on three dimensions. We consider this a preliminary classification that is likely to evolve with further research. The X dimension considers whether events are recurrent or rare. Recurrent context events result in patterns describing repetitive activities in contrast with rare events that are anomalies or deviations from normal activities. Any of the above examples from Section 1 are applications that fall in the recurrent event class. Examples of CCA applications involving rare events include intrusion detection and credit card fraud applications that leverage context to detect deviation from normal behaviors and predict anomalies. The Y dimension addresses the relationship between events within a pattern. A pattern that simply treats its events as the elements of a set falls into the category of grouped event. Patterns in which events are partially ordered fall in the ordered event category. If events are totally ordered and the absolute start time of each event is important, then the appropriate category is timed event. The Z dimension considers the length of each event. If an event is conceptually instantaneous

(e.g., a web page request) then its position in the Z dimension is instantaneous. If the event has a nonzero duration (e.g., "going home"), then it falls within the duration category along the Z dimension.

Below are examples of customized context-aware applications covering several sections of our application space.

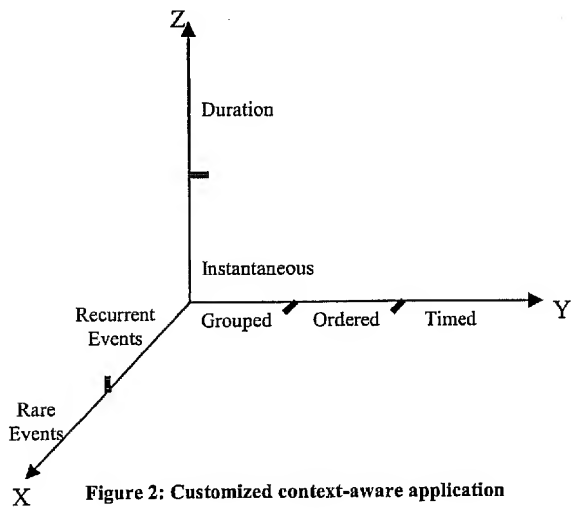

Figure 2: Customized context-aware application

- *Context-Aware Content Distribution (CACD)*: The goal of the CACD application is to predict web pages that a user plans to access and then pre-fetch and pre-transcode the pages to reduce user-observed download latency. In this case, context includes the device display type (e.g., desktop PC, palm pilot), the user location, and the web page access history which are used to infer the next page a user will access. We have already developed an initial prototype of this application that mines only web access logs and pre-fetches web content to an Apache web prefetching module [7].

- *Context-Sensitive Scheduler:* The goal of this application is to schedule computationally expensive utilities (such as file backup and antivirus utilities) dynamically to run on a laptop at times convenient for the owner. The Context-Sensitive Scheduler utility uses context about a laptop's state as well as location and calendar information about the owner to infer that the laptop will be idle long enough to accommodate the execution of the relevant task. Thus, CSS considers timed events that have duration.

- *Smart HVAC System*: The goal of the Smart HVAC system is to automatically adjust the thermostat of a person's house based on whether the house is occupied. The system uses diverse forms of context that include the home owner's location and calendar to infer that the home owner is headed home and the thermostat temperature should be increased or decreased to a comfortable temperature. Thus, this application has timed events that are conceptually instantaneous.

- *Credit-Card Fraud Detection:* Context can be used in very simple ways to determine credit card fraud; e.g., if a person's location indicates that they are in Dallas while their credit card is being used in Chicago then signal an alarm. Beyond such simple uses, patterns of credit card use can be determined and deviations from these patterns can be used to signal that fraudulent activity is occurring. Such rare events are likely to require particular machine learning techniques that are quite different from the techniques of other applications in our classification scheme.

- *Clustered Event Scheduler:* The occurrence of related events is often stimulated by context. E.g., prior to a business trip one must book a flight as well as reserve a rental car and hotel room. Currently the correlation of these activities (flight, car and hotel web-based reservations) is manually determined. We suggest that learning techniques can be used to infer the relationship between these events as an example of grouped events within our classification.

4. Challenge 1: A Programming Model

Our fundamental premise for the design of a programming model is that a context-aware application can be modeled as a set of services in which each service consists of two separate functions: triggering and effecting. The *triggering* function interprets context to determine if the service should initiate, pause, resume, or cease execution. For example, the Context Sensitive Scheduler interprets context (e.g. location and calendar context) to infer computer idleness and determine whether computationally intensive applications can be triggered. Triggering functionality is highly dependent on the user's behavior, but usually independent of the application's goal. For example, the trigger that infers that a frequent traveler is headed to their temporary residence varies significantly based on the traveler's habits, but the decision is applicable to a variety of services (e.g., temperature and lighting, food services). The *effecting* function performs actions based on the trigger. The effecting functionality can be broadly applied across many users, but is tightly coupled with the application's goal (e.g., controlling the temperature in a hotel room based on multiple sources of context such as the internal and external temperature, the state of all windows and the state of ceiling fans).

The decoupling of triggering and effecting functions lies at the heart of our approach. With this decoupling principle,

we move most of the triggering operations from the application space and place them into the supporting framework. We note two benefits to this approach. From a general system design perspective, this principle exploits the reusable nature of triggering operations across applications. More importantly, from an application development perspective, this principle will dramatically simplify the development of context-aware applications by allowing application developers to focus on the design of a minimal set of tasks mainly defined by the effecting component of their application.

One challenge that we face in designing our programming model is to design a generic API that application developers can use to interface the effecting aspects of their applications with a framework that supports the triggering aspects. This API should be as simple as possible and should not assume or require that application developers be familiar with the underlying machine-learning algorithms used to perform triggering operations.

4.1. High Level Architecture

A preliminary high level architecture of the middleware supporting our programming model is shown in Figure 3. This architecture is built on top of an existing context service that we have already designed, as discussed in [3]. It consists of this context service with a set of context sources, a context log repository, a learning engine, a context pattern repository and a pattern activator. In addition a vocabulary is defined and associated with the context service. The context service acts as a provider of context information. It contains all the functionality needed to dispatch relevant context from context sources to its clients. We attach to the context service a vocabulary describing each context source and semantics of the data that is provided. Examples of context sources are location, calendar, current user activity and network accessibility.

All requests for context are logged by the context service in a context log repository. The format of individual log entries consists of four fields: a time stamp, a user identification, a context type, and a context state. The time stamp logs the time when the request for context was served by the context service. The user identification field logs the subject on behalf of who the request was made. The context type field logs the context source used to satisfy the request. The range of values of a context type must be expressed in the vocabulary associated with the context service. Finally, the context state field logs the value taken by the context type at a given time, for a given user.

Using context logs, the learning engine derives context patterns and stores them in the context pattern repository. The derivation of these patterns is performed by machine learning algorithms operating on the context logs. The output of the learning engine is a set of context patterns. These patterns have two parts: a condition and an action. The condition portion defines the state in which various context attributes must be for the pattern to be activated. The action portion is a trigger to applications. The learning engine appends to these patterns statistical properties measuring the accuracy of the derivation. For example, the learning engine could determine that 95% of the time, when user A drives past a given location on Main Street, user A will arrive to his hotel, within 25 minutes. The learning engine stores all these results in the context pattern repository.

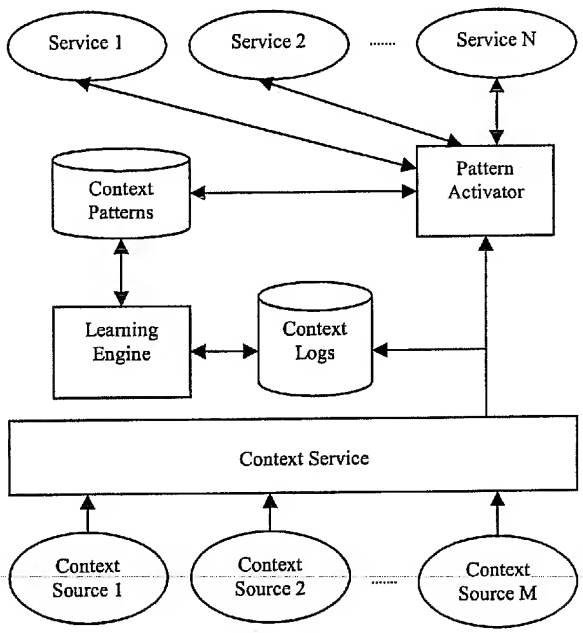

Figure 3: High Level Architecture

The pattern activator accesses the context pattern repository and triggers the effecting part of context-aware applications that are shown in Figure 3 as services. To perform this task, it allows applications to set up subscriptions for context triggers. The vocabulary of context is made available to application developers to determine the right triggers needed by their applications and set up the appropriate subscriptions. To serve these subscriptions, the pattern activator accesses current context from the context service and attempts to match it with the condition part of relevant context patterns obtained from the context repository. This matching step generates a set of context patterns that are used to activate services. Only the action parts of these patterns are retained and used to serve applications subscriptions. Accordingly, the format of the subscriptions sent by applications to the pattern activator is composed of triggers with a required accuracy and a required time window.

4.2. Refining the Architecture

Related to the architecture is the need to support feedback from the applications to the middleware about the performance of the triggering operations. The performance of the triggering operations must be measured with generic application metrics. To use these measures and improve the performance of triggering operations, these generic metrics must be automatically mapped into another set of performance metrics specific to the triggering operations used inside the framework.

To illustrate this problem, consider the CACD application. The effecting part of CACD resides in an apache proxy module that we have designed to pre-fetch content in a proxy cache close to the end user. In this portion of the application, the performance of the application is naturally measured by the hit rate improvement that pre-fetching adds to pure caching techniques. The decision on what to pre-fetch is made by an external pre-fetching server that contains the triggering portion of the application. In this server, the performance is measured by the prediction accuracy of the learning scheme used. There is a complex relationship that exists between the hit rate increase measured at the application layer and the prediction accuracy measured within the middleware. To close the loop and adjust the triggering operations to adapt to changes at the effecting layer, it is imperative to understand this complex relationship. Initially, we propose that the application developer explicitly specifies this mapping. As we build more applications, we will identify the common parts of the mapping that applications share and automate these parts in the framework.

5. Challenge 2: An Inference Framework

The different classes of customized context-aware applications have different learning requirements. For our framework to be effective we must ensure that it contains a set of efficient learning algorithms for each segment of the classification. As a result, we face the challenge of designing a machine learning toolkit for customized context-aware applications.

For the ordered event class of applications, we are extending the initial work that we have done for the Context Aware Content Distribution Application [7]. In that work we developed an algorithm that can be extended to the entire ordered event class. This algorithm, called Fuzzy-LZ, is an extension of the parsing algorithm used in Lempel-Ziv compression. Fuzzy-LZ is a sequential predictor that makes approximate URL access predictions from past URL access. The approach is "fuzzy" because it ignores the small variations in the structure of the URLs composing the input sequence. In essence, it groups similar URLs into clusters and instead of predicting individual URLs, it predicts clusters of URLs. This approach allows us to take into account the fact that users access similar, but not necessarily identical, Web URLs. For example, all the news stories located at CNN.com related to the US Open tennis tournament on September 2 and September 3, 2003 might have similar but not identical URLs on a news web site. This group of similar URLs is treated by our algorithm as a single unit. A metric used to express closeness of URLs is defined on general parts of URLs such as the server name, path name and file name. While we have developed Fuzzy LZ for web mining problems, we believe that it can be extended to other forms of context. This extension requires the definition of metrics for such forms of context that will allow us to extract meaningful patterns from context logs by ignoring unnecessary details.

Applications in the grouped event class have learning requirements that can be met with unsupervised learning techniques. Unsupervised learning techniques form clusters or "natural groupings" of the input patterns [12]. In each cluster, the order of occurrence of each event is not important. The grouping criteria used to produce these clusters are arbitrary. A typical criterion is the event time stamp that allows the grouping of context events that tend to occur simultaneously. For instance, one could learn from logs of context that whenever a user arrives at work, she always synchronizes her mail client with her mail server, accesses the same portal web site, checks her calendar and gets coffee from the cafeteria, but not necessarily in this order. Hence, whenever she gets coffee, the other applications in this set may benefit from this knowledge. For example, if she has not accessed her mail yet, mail replication could be done on her favorite devices. We are currently developing such unsupervised techniques to mine contextual data.

We are also developing learning algorithms for the timed event class of applications. Such algorithms have received little attention in the literature. One exception is the work of Ma and Hellerstein [4] where the authors proposed a clever algorithm that mines partially periodic patterns with unknown periods. Unfortunately, this algorithm does not address key aspects of the learning requirements for timed event applications. For instance, it was not designed to predict that a given context event will occur in T time units. Furthermore, their technique was not designed to determine the duration of a given context event. Hence, it does not solve the learning problems posed by the Context-Sensitive Scheduler and Smart HVAC system applications. Our initial approach extends the current state of the art in machine learning practices to such time prediction problems.

Learning algorithms for rare events face different challenges from the ones addressed by all the problems presented above. Indeed, the challenge that we face here is the design of techniques able to predict irregularities with a small amount historical data. A typical application in this space involves the prediction of fraudulent credit card transactions. In such examples, we are looking for deviations from normal behaviors to trigger alarms. An obvious approach to solve this problem is to actually mine for recurrent patterns and trigger alarms when events do not match any of these recurrent patterns.

6. Challenge 3: Security and Privacy

Security and privacy are important to most customized context-aware computing applications. Context that is available about a given user often has a sensitive nature. For example, knowing that a person's location is far from their home can be used to a thief's advantage. Users need controls that restrict context to only be used how and when the user wants. Any context-aware application must have facilities for controlling context to satisfy these requirements.

A context-aware application that depends on context from a single domain can assume that the domain is a trusted environment in which users and applications are easily authenticated. Any application that depends on context from multiple domains does not have this luxury and must be authenticated to multiple domains. This may lead to a world in which users are forced to authenticate applications numerous times resulting in an intractable user ID/password management problem.

A major privacy challenge with context-aware application development involves the fact that many applications use context that is owned by several distinct organizations. The organizations or context domains have administrative control over the collection and dissemination of context on behalf of a given user within the confines of the domain. An example context domain might be owned by a cell phone service provider that generates location context about a user based on the state and location of the user's cell phone; a corporation might own a context domain that generates context about its employees while at work as well as other general forms of context.

The Context Tailor infrastructure as well as any context service that uses an inference framework to derive patterns about user context introduces a new class of assets about which privacy matters. In addition to user context, user's context patterns themselves may require their own privacy policies. E.g., a user's location may be private; likewise, the pattern about a user's location every Tuesday at 3 PM may be private as well. The introduction of context patterns as a new asset to be considered with respect to privacy raises the stakes on how secure an infrastructure involving Context Tailor should be. In other words, breaches of security for Context Tailor compromise both the sources of context that Context Tailor monitors as well as the patterns that Context Tailor derives.

The above privacy and security concerns are common among many context infrastructures beyond the Context Tailor project. The Context Tailor infrastructure adds an extra privacy burden over and above typical context-aware applications with respect to a user's specification of privacy constraints. Generally, Context Tailor requires a user to specify more privacy constraints for a given application than would be the case if a context-aware application did not leverage our infrastructure.

To understand this point, consider the fact that a context-aware application can be characterized as a function of N context sources where each source of context has M states. The complexity of specifying privacy constraints for such applications is generally $O(M^N)$. A user of a context-aware application must specify privacy constraints for each application about each relevant source of context. Consider k applications written without our infrastructure such that their separate machine learning facilities each depend on N distinct sources of context. Specifying the complexity of these k sources of context is $O(kM^N)$. If the applications are written using the Context Tailor infrastructure the complexity of specifying privacy becomes $O(kM^{kN})$ since each application will now depend on all of the context sources available through the context service.

7. Concluding Remarks

In this paper we have presented three challenges central to the design of an application model for CCA applications: the design of a programming model supporting the decoupling of effecting and triggering functionalities, the design of a machine learning toolkit that supports triggering functions and an approach to protect the security and privacy of users. While we are currently addressing each of these challenges, there are two additional issues that we plan to address in the future, as the scale of the system grows.

First, our framework and programming model will only be successful if it can accommodate a large, heterogeneous set of context sources. Each context source is provided with a name and several attribute-value pairs that describe the state of context events. As the number and type of context sources grow there will be a great deal of redundancy and overlap of context types. Indeed, our existing context service has multiple notions of location context: a Blackberry

server that indicates location based on nearest cell tower as well as an 802.11 location server. The naming scheme distinguishes these two types of location context even though conceptually we would like to treat these forms of context as being similar. Consider the case of a context pattern consisting of Blackberry location events that is matched to a particular action, e.g., adjust the thermostat. If a new 802.11 event occurs that is located near the Blackberry location pattern, we would like the Context Tailor framework to trigger the same thermostat action. The challenge in this problem is to develop an extensible notion of closeness between forms of context that are syntactically different but semantically similar. Several ontological engineering research endeavors are currently exploring similar goals such as the Resource Description Framework (RDF) and the DARPA Agent Modeling Language (DAML) including many sub-disciplines of DAML. We will leverage these efforts together to tackle this problem.

Second, as the number of generated context events grows a need for computational efficiency will arise to accommodate the large set of stored context patterns. To define this problem, consider the sequence of steps involved during runtime when the framework attempts to decide which applications should be triggered. As new context from the user is obtained from the underlying context infrastructure, the framework needs efficient mechanisms to match this incoming context with the list of patterns generated by the learning toolkit. The computational complexity of this pattern matching is proportional to the number of applications times the number of patterns times the number of users.

For some timed event applications, hard deadlines defining when the applications should be triggered may be specified by the developer. As a result, it is imperative to develop efficient pattern matching techniques. Our initial approach to this problem will make use of state of the art matching techniques such as the Rete Algorithm [16] to match incoming context events to patterns that were learned. We will extend these techniques to take into account the diverse application triggering requirements following a general rule of thumb that first looks for valid patterns for applications with strict deadlines and then tries to identify patterns for other applications that can tolerate triggering delays.

Acknowledgement: The authors would like to warmly thank Dr. Guruduth Banavar for all the comments and suggestions that he has made on this work.

References

[1] Norman Cohen, Hui Lei, Paul Castro, John S. Davis II, Apratim Purakayastha, "Composing Pervasive Data with iQL," Proceedings of the 4th *Annual IEEE Workshop on Mobile Computing Systems & Applications*, Callicoon, NY, June 2002, pp 94-104.

[2] Jason I. Hong and James A. Landay, "An Infrastructure Approach to Context-Aware Computing." In *Human-Computer Interaction*, 2001, vol. 16.

[3] Hui Lei, Daby Sow, John S. Davis II, Guruduth Banavar, Maria Ebling, "The Design and Applications of a Context Service," Mobile Computing and Communications Review, to appear.

[4] Sheng Ma and Joseph L. Hellerstein, "Mining Partially Periodic Event Patterns with Unknown Periods", Proceedings of the 17th International Conference on Data Engineering, April 2-6, 2001, Heidelberg, p 205-214

[5] D. Salber, A.K. Dey, and G.D. Abowd, "The Context Toolkit: Aiding the Development of Context-Enabled Applications," In Proceedings of ACM CHI99, Pittsburgh, PA, pages 434–441, May 1999.

[6] M. Satyanarayanan, "Pervasive Computing: Vision and Challenges," *IEEE Personal Communications*, August 2001, pp 10-17.

[7] Daby Sow, David P. Olshefski, Mandis Beigi and Guruduth Banavar, "Prefetching based on Web Usage Mining", Proceedings of ACM/IFIP/USENIX Middleware 2003, to appear.

[8] Roy Want, Bill N. Schilit, Norman I. Adams, Rich Gold, et al, "An Overview of the PARCTAB Ubiquitous Computing Experiment," *IEEE Personal Communications*, December 1995, pp 28-43.

[9] Mark Weiser, "The Computer for the 21st Century," *Scientific American*, vol. 265, no. 3, September 1991, pp 66-75.

[10] M. Weiser and J. Brown. *Designing calm technology*. PowerGrid Journal, 1(1), 1996.

[11] Stephen S. Yau, Fariaz Karim, Yu Wang, Bin Wang and Sandeep K.S. Gupta, "Reconfigurable Context-Sensitive Middleware for Pervasive Computing," *IEEE Pervasive Computing*, vol. 1, no. 3, July – September 2002, pp 33-40.

[12] Duda, Hart and Stork, Pattern Classification, 2nd edition, Wiley Interscience, 2002.

[13] Jeremy Goecks and Jude Shavlik, "Learning Users' Interests by Unobtrusively Observing Their Normal Behavior," Proceedings of the *International Conference on Intelligent User Interfaces*, pp 129-132, 2000.

[14] Cecilia Mascolo, Licia Capra, and Wolfgang Emmerich, "Mobile Computing Middleware," Lecture Notes In Computer Science, vol. 2497, Springer, pp 20-58, 2002.

[15] S. K. Das, D. J. Cook, A. Bhattacharya, E. O. Heierman, III, and T.-Y. Lin, "The Role of Prediction Algorithms in the MavHome Smart Home Architecture," *IEEE Wireless Communications Communications Special Issue on Smart Homes*, 9(6), pages 77-84, 2002.

[16] S. Russel and P. Norvig, Artificial Intelligence: A Modern Approach, Prentic-Hall, 1995.

A Network Authentication System with Multi-Biometrics

Shoichiro Seno, Tetsuo Sadakane, Yoshimasa Baba and Toshihiro Shikama
Information Technology R&D Center, Mitsubishi Electric Corporation, Kamakura, 247-8501 Japan.

Yuuji Koui and Naoshi Nakaya
Faculty of Engineering
Iwate University
Morioka, 020-8551 Japan.

Abstract—**In view of the recent increase of incidents over the Internet and other networks, the role of authentication techniques to prevent unauthorized access by malicious users becomes more significant. User authentication methods can be classified into three categories: (1) methods based on human memory such as passwords, (2) methods based on physical devices such as magnetic or IC cards, and (3) methods based on biometrics such as fingerprint and iris. As the first two categories cannot escape vulnerabilities caused by forgetfulness or losses, the third category attracts much attention in these days. This paper proposes a network authentication system with multi-biometrics to support various applications where user authentication is necessary. In particular, it can provide authentication services to a workflow process that is a typical intranet application. The paper also discusses the prototype implementations developed after the proposed system and their evaluation.**

Keywords-biometrics; network authentication system; workflow

I. INTRODUCTION

In the modern society, all kinds of public and private services are more or less dependant on computer networks supporting them. Electronic voting and electronic commerce are two main examples of them. Because crimes and incidents over networks are increasing rapidly, the role of authentication techniques to prevent unauthorized access by malicious users becomes more significant.

User authentication methods currently in use can be classified into three categories: (1) methods based on human memory such as passwords or Personal ID, (2) methods based on physical devices such as magnetic or IC cards, and (3) methods based on biometrics such as fingerprint and iris. The methods in the first two categories are inherently vulnerable due to forgetfulness and physical losses. As such, the use of methods in the third category is becoming more popular.

Biometrics authentication depends on biological individuality of human characteristics such as fingerprint, iris, retina, face, and voice. A biometrics authentication technology automatically extracts identification data from such human characteristics and compares it with pre-registered data to authenticate a person, but how it is done is different according to the characteristics it focuses. Biometrics authentication has three advantages: (1) difficulty of reproduction, (2) safety from theft, and (3) ease of use (no needs to remember passwords or keep an ID card). These advantages help its wider use in recent years, and it can be expected that it will become much popular with the advance of biometrics authentication technologies to produce inexpensive and small devices to process biometrics data.

Biometrical authentication technologies differ each other with respect to preciseness, device size, cost, and appropriate application areas. Also, user acceptance of biometrical authentication should be considered, as it differs on the human characteristics being used. Some users may oppose use of their fingerprint.

A major problem of biometrics authentication is that it is not free from an error in the process of extraction of human characteristics and comparison of biometrics data. Multi-biometrics is useful to improve reliability of biometrics authentication when a single biometrics authentication technology cannot satisfy a required reliability level. For example, fingerprint authentication at the entrance of a building may be combined with iris authentication at the entrance of a secured room in that building.

Because biometrics authentication will be more popular over networks in the future, it is useful to build a network-based biometrics authentication platform for use by many applications and commonly applicable to different types of biometrics authentication technologies, including combination of them (i.e., multi-biometrics). This paper proposes a network authentication system with multi-biometrics to support various applications where user authentication is necessary. In particular, it can provide authentication services to a workflow process which schedules jobs to responsible persons according to a specified sequence.

In the following, chapter II reviews past researches of biometrics authentication technologies and considers how they can be used for network applications. Chapter III proposes basic design criteria of a network authentication system. Chapter IV presents the design based upon the proposed criteria, and Chapter V discusses implementation and evaluation of the prototypes developed after the design. Chapter VI is the conclusion.

II. BIOMETRICS AUTHENTICATION TECHNOLOGIES

A. *Past Biometrics Authentication Researches*

Past researches are summarized as follows:

1) Researches to Recognize Biometrical Characteristics: Researches on extraction of biometrical characteristics of a human began in 1960s. Based on them, FBI adapted fingerprint recognition for crime investigation in 1980s [1]. Meanwhile, researchers examined other types of biometrics and other applications. In 1990s, software development eased

0-7803-8114-9/03/$17.00 ©2003 IEEE.

implementation of extraction techniques, which paved the way for deployment of biometrics authentication [2]. Today researchers are studying fingerprint [3][4][5][6][7][8], iris [9][10], retina [11], hand geometry [12][13], voice [14][15], and face [16][17] for sources of information to uniquely identify a human. Also, there are studies on movement of the pen and/or fingers to make a signature for the same purpose [18][19][20].

2) Researches to Establish Biometrics Authentication: Some researchers considered common issues with extraction of identification data from various types of biometrics, and protection of such data against conceivable attacks [21][22][23]. They aimed at facilitating reliable biometrics authentication by improving authentication preciseness and providing counter-measures against attacks to an authentication system.

3) Researches to Combine Multiple Biometrics Authentication (Multi-biometrics): It is possible to strictly authenticate a person by combining multiple biometrics authentication methods while accepting some degree of authentication failures with a single biometrics. Combination of biometrics authentication [24] may be achieved by logical or statistical methods. Logical methods perform each of biometrics authentication individually and take AND or OR of their results to reach the final answer. Statistical methods rely on a statistical function derived by matching probabilities by individual authentication methods.

B. *Biometrics Authentication over Networks*

For biometrics authentication to be used reliably over the Internet and other networks, the whole process of authentication must be well protected. In a network authentication system shown in Fig. 1, biometrics information extracted by the sensor device is relayed to the authentication server by the authentication client over a network. Because information sent over a network is subject to eavesdropping and forgery, each step taken by the authentication system shall be protected from conceivable threats by a security architecture encompassing all involved elements. For example, such an architecture should take into the account confidentiality and integrity of authentication information over a network, security measures to protect involved elements, and considerations on reliability of biometrics authentication. It is worth noting that careful implementation is also necessary to ensure security of the whole system once the security architecture is defined. Whereas most of the above dealt with improvement of biometrics authentication technologies, we have been working on design and implementation issues of a biometrics authentication system over a network [25].

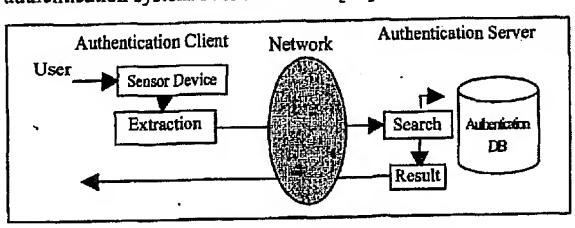

Figure 1. A Network Authentication System

C. *Use of Biometrics Authentication by Network Applications*

We consider typical use of biometrics authentication over a network to be enhancement of authentication accuracy for Web-based user interfaces because of their popularity over the Internet and intranets.

In our previous works, a biometrics authentication system over a network which can be integrated easily with a Web-based application is presented [25]. This paper proposes an extended version of the biometrics authentication system supporting multi-biometrics and integration capability with workflow processes. A workflow process is a computer-aided scheduling of jobs widely used over an intranet. Its examples include an approval process of a report from an employee to his/her supervisor, and an electronic procurement process.

III. Basic Design Criteria of a Network Authentication System

We have identified the following basic design criteria for a network authentication system with multi-biometrics. They are assumed as a guideline for design and implementation described in the later sections.

(1) The authentication system must support multiple biometrics as well as password-based authentication, and any combination of them by an individual user. Also it must support not only verification but also identification by biometrics.

(2) It must provide centralized management of authentication data and related information such as personal data and access control conditions, in order to minimize management overhead to maintain such information.

(3) It must provide an authentication interface to Web-based applications for use of biometrics.

(4) It shall use authentication protocols adequately defined to defend authentication information sent over networks from possible loss and conceivable attacks such as impersonation and replay.

(5) It shall provide integration capability with workflow processes.

IV. Network Authentication System Design

A. *Network Authentication Models*

We have identified two authentication models as key authentication architectures over networks.

1) The Co-located Model: The Co-located model assumes that a service requiring user authentication is physically tied with a user's location, as shown in an example in Fig. 2. In this example, when a user requests the gatekeeper to open the door, it obtains biometrics data from the user by means of the sensor device and requests the Authentication Client to verify the user by sending an authentication request to the remote Authentication Server. Then the Authentication Server sends back a response to admit or to deny opening of the door.

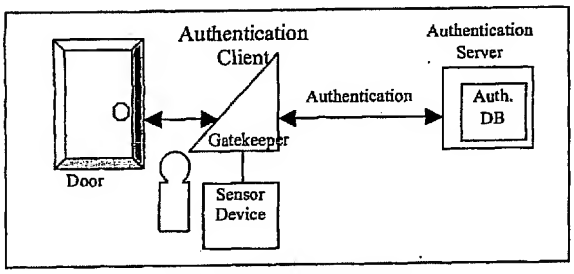

Figure 2. The Co-located Model

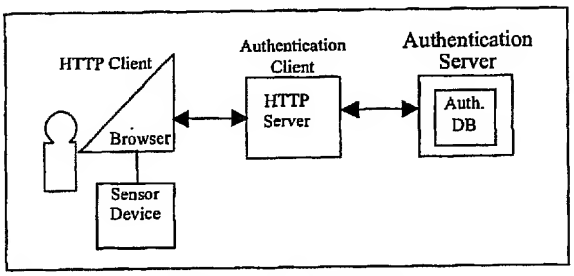

Figure 3. The Separated Model

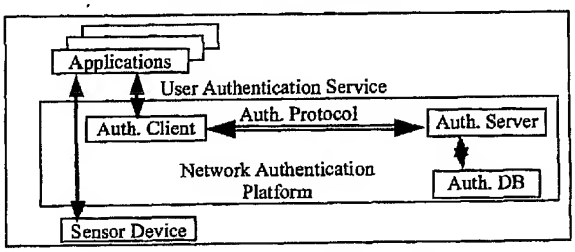

Figure 4. A Network Authentication Platform

2) The Separated Model: The Separated Model assumes that a service requiring user authentication is physically separated from a user's location, as shown in an example in Fig. 3. In this example, when a user requests access to a content in the HTTP(Hypertext Transfer Protocol) server, the HTTP server requests the HTTP client (browser) to obtain biometrics data from the user by means of the sensor device and send it. When it receives the biometrics data, the HTTP server requests the Authentication Client to send an authentication request including the data and the content's ID to the remote Authentication Server via CGI(Common Gateway Interface). Then the Authentication Server sends back a response including the user's access privilege to the content.

B. *A Network Authentication Platform*

We have developed a network authentication platform [26] upon which authentication services can be built according to either the Co-located Model or the Separated Model. As shown in Fig. 4, the network authentication platform consists of the following software components: the Authentication Client, the Authentication Server, and the Authentication Database. The Authentication Client provides user authentication service to applications based on biometrics data extracted by the sensor device. The Authentication Server receives authentication requests with biometrics data from the Authentication Clients and responds by matching the data with the Authentication Database. The Authentication Database stores user biometrics data and other data such as personal information of users and access control conditions for individual applications. These software components directly map to the elements of Co-located Model. In the case of the Separated Model, extraction of biometrics characteristics of a user can be delegated to an agent residing with the browser. Extracted biometrics data may be communicated from the agent to the Authentication Client by a secure protocol, e.g., SSL(Secure Socket Layer) [26]. As for the authentication protocol between the Authentication Clients and the Authentication Server, the platform implemented RADIUS [27] with an enhancement to securely transport biometrics data.

C. *Support of Multi-biometrics*

Length of data used for biometrics authentication is typically in the range of a few to one and half thousand Bytes. Such data may be extracted by a sensor device from fingerprint, iris, and other biometrics and stored in the Authentication Database in the Authentication Server. The Authentication Database can be configured to support multi-biometrics by allocating multiple fields to store individual biometrics data. The same type of biometrics data may occupy multiple fields to improve preciseness or to store back-up data in case of injury (e.g., data extracted from another finger for fingerprint authentication).

D. *Management of the Authentication Database*

The Authentication Database shall include personal information corresponding to stored biometrics data and access control information to authorize access to Applications based on a user's identity. Access control may be combined with multi-biometrics, e.g., to mandate authentication by multiple biometrics to use an application requiring high security.

Table I shows an example user file structure in the Authentication Database. The password field is defined in the structure for the following purposes:

(1) To provide alternative authentication means for users from whom extraction of biometrics data is technically difficult.

(2) To authenticate a user for initial registration of biometrics data.

(2) in the above allows a user to register biometrics data remotely from his/her site over a network, thus reducing overheads with registration and maintenance of biometrics data in the Authentication Database. For initial registration of biometrics data, the administrator issues a temporary password with which only registration of biometrics data is allowed. Fig. 5 shows an example initial registration procedure. The registration procedure will be protected from impersonation over the network by secure communication protocols among participants of the procedure, i.e., SSL and RADIUS in the case of our platform.

TABLE I. AN EXAMPLE USER FILE STRUCTURE

User ID	Status	User Info	User Type	Biometrics Type 1	Biometrics Type 2	Password	Access Control	Validity
User-1	Status-1	Info-1	T1	Finger-1-1	Finger-1-2	Passwd-1	A1, A2	
User-2	Status-2	Info-2	T2	Iris-2-1	Finger-2-1	Passwd-2	A2, A4	
User-m	Status-m	Info-m	T2	Rétina-m-1	Finger-m-1	Passwd-m	A1, A3	

Status: Waiting for registration/Access permitted/Access denied),

User Info: Name, affiliation, telephone no., etc., User Type: Administrator/User

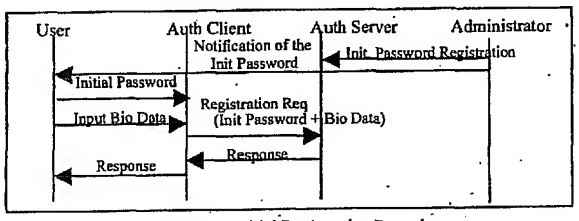

Figure 5. Initial Registration Procedure

E. Integration with a Workflow Process

A workflow process provides scheduling of jobs to responsible persons usually through the Web interface and Emails. For example in a report approval process, an employee starts a workflow process by submitting a report electronically. Then the workflow process requests approval of the employee's supervisor by an Email, and the supervisor reads the report and issues approval through the Web interface.

Security of a workflow process will be greatly enhanced if biometrics authentication of responsible persons is integrated within the process. This can be accomplished by the use of the network authentication platform for the Separated Model with the following features: (1) Biometrics authentication means for each person who takes part in a workflow process; and (2) Storage of authentication schedules according to workflow processes in the Authentication Database.

In the authentication scheduling according to a workflow process shown in Fig. 6, a triggered workflow process, Authentication Client 1, requests authentication of User 1and User 3, respectively, to reach completion of the workflow process denoted by Application 1. Alternative users, i.e., User 2 and User 3 for User 1, and User 4 for User 3, and alternative applications are also registered to deal with exceptional handling.

F. Protection of Biometrics Data over a Network

Our platform chose SSL and RADIUS for the protocol to transfer biometrics data between the agent and the Authentication Client, and the Authentication Client and the Authentication Server, respectively. SSL can adequately protect biometrics data sent over a network by a standard encryption algorithm. As for RADIUS, it only supports encryption of passwords that are much shorter than usual biometrics data. Our platform enhanced RADIUS by adding encrypted transfer of multiple types of large identification data, including user ID to suit for multi-biometrics.

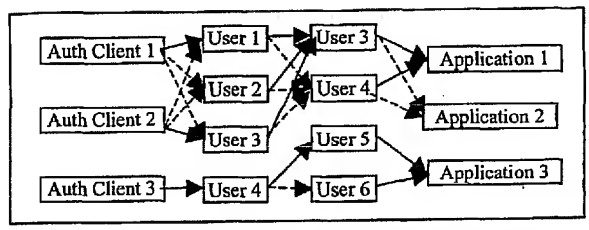

Figure 6. Authentication Scheduling according to Workflow Processes

V. IMPLEMENTATION AND EVALUATION

Two prototype implementations, called Type A and Type B here, have been developed after the above described network authentication platform adapted to the Separated Model. In Type A, the Authentication Client and the Authentication Server were built upon Solaris Operating System. In Type B, they were built upon Windows NT. Another difference between them is the degree of optimization of the software codes, in particular with the CGI interface and Authentication Database access methods, of which Type B is better. More details of Type A are described in [25].

A. Measurement of Processing Time

Both implementations were evaluated with respect to relative processing time of the software components in the authentication sequence shown in Fig. 7. In it, the dotted lines indicate time intervals that were measured. The information exchanged between the components were as follows:

a: the user request including the user ID and biometrics data (1K Bytes)

b: the authentication request including biometrics data

c: the authentication response indicating OK/NG

d: the content requested by the user (an HTML file)

Fig. 8 and Fig. 9 show measurement results of relative component processing time by Type A and by Type B, respectively. The total processing time was 1.5 sec by Type A and 0.4 sec by Type B. In both cases, processing time of the authentication protocol, RADIUS, is smaller than other components. It can be concluded from these results that processing time of biometrics authentication over a network is not an obstacle to its deployment, especially when its implementation is well optimized.

B. Applicability of the Implementation to Multi-Biometrics

Although we measured processing time of the implementation only in the case of fingerprint authentication with 100 entries in the Authentication Database, we believe that relative smallness of time required for transport of biometrics data over a network will hold true with other types of biometrics. This is because the length of biometrics data is usually less 1.5 kilobytes after extraction regardless of types of biometrics. The results in Fig. 8 and Fig. 9 suggest that a larger size of biometrics data may not enlarge authentication time over a network very much.

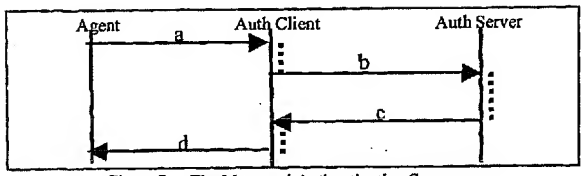

Figure 7. The Measrued Authentication Sequence

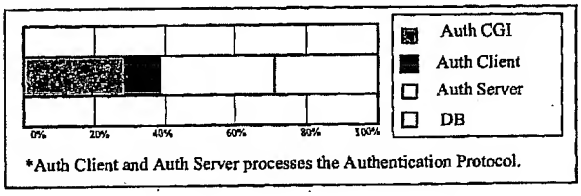

Figure 8. Ratio of Processing Time by Type A

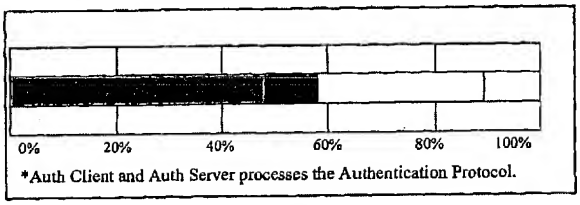

Figure 9. Ratio of Processing Time by Type B

VI. CONCLUSION

This paper proposes a network authentication system with multi-biometrics to support various applications where user authentication is necessary. In particular, it can provide authentication services to a workflow process which schedules jobs to responsible persons. The paper also discusses prototype implementations developed after the proposed network authentication system and their evaluation.

It was shown that the proposed system would be useful to build a network authentication system with multi-biometrics, with sufficiently small authentication processing time and wide applicability to network applications, including these corresponding to the Co-located Model and the Separated Model.

Further study items include implementation and evaluation of biometrics other than fingerprint, and efficient management of the Authentication Database to enable application-specific authentication and to better support workflow processes.

REFERENCES

[1] D. W. Davies and W. L. Price, "Security for Computer Networks," pp.169-208, John Wiley & Sons, 1986.

[2] B. Miller, "Vital Signs of Identity," IEEE SPECTRUM, Vol.2, pp.22-30, 1994.

[3] A. K. Jain, R. Bolle, and S. Pankanti (Eds.), "BIOMETRICS: Personal Identification in Networked Society," Kluwer Academic Publishers, 1999.

[4] R. Germain, A. Califano, and S. Colville, "Fingerprint Matching Using Transformation Parameter Clustering," IEEE Computational Science and Engineering, Vol.4, No.4, pp.42-49, 1997.

[5] A. K. Jain, S. Prabhakar, and L. Hong, "A Multichannel Approach to Fingerprint Classification," IEEE Trans. on PAMI, Vol.21, No.4, pp.348-359, April 1999.

[6] A. K. Jain, S. Prabhakar, and S. Chen, "Combining Multiple Matchers for a High Security Fingerprint Verification System," Pattern Recognition Letters, Vol.20, No.11-13, pp.1371-1379, 1999.

[7] S. Shigematsu, H. Morimura, Y. Tanabe, T. Adachi, and K. Machida, "A Single-Chip Fingerprint Sensor and Identifier," IEEE Jour. Solid-state Circuits, Vol.34, No.12, pp.1852-1859, December 1999.

[8] K. Uchida, "Fingerprint Identification for Enhanced User Interface and for Secure Internet Services," IEICE Trans. Inf. and Syst., Vol.E84-D, No.7, pp.806-811, July 2001.

[9] P.W. Hallinan, "Recognizing Human Eyes," SPIE Proc. Geometric Methods in Computer Vision, 1570, pp.214-226, 1991.

[10] S. Lim, K. Lee, O. Byeon, and T. Kim, "Efficient Iris Recognition through Improvement of Feature Vector and Classifier," ETRI Journal, Vol.23, No.2, pp.61-69, June 2001.

[11] R. Hill, "Retina Identification," BIOMETRICS: Personal Identification in Networked Society [4].

[12] A. K. Jain, A. Ross, and S. Pankanti, "A Prototype Hand Geometry-Based Verification System," in 2nd International Conference on Audio- and Video-based Biometric Person Authentication, Washington D.C., March 1999.

[13] H. Yusa, A. Hyogo, and K. Sekine, "A Method of Hand Shape Recognition Extracting Datums on 2 Dimensional Plane," Trans. IEICE, Vol.J80-D-II, No.5, pp.1209-1220, May 1997 (in Japanese).

[14] M. H. George and R. A. King, "A Robust Speaker Verification Biometric," Proc. IEEE 29th Annual 1995 International Carnahan Conference On Security Technology, pp.41-46, UK, October 1995.

[15] C. Miyajima, Y. Hattori, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Text-Independent Speaker Identification Using Gaussian Mixture Models Based on Multi-Space Probability Distribution," IEICE Trans. Inf. and Syst., Vol.E84-D, No.7, pp.847-855, July 2001.

[16] J. Weng and D.L. Swets, "Face Recognition," BIOMETRICS: Personal Identification in Networked Society [4].

[17] K. Hotta, T. Mishima, and T. Kurita, "Scale Invariant Face Detection and Classification Method Using Shift Invariant Features Extracted from Log-Polar Image," IEICE Trans. Inf. and Syst., Vol.E84-D, No.7, pp.867-878, July 2001.

[18] Y. Yamazaki and N. Komatsu, "A Proposal for a Text Indicated Writer Verification Method," IEICE Trans. Fundamentals, Vol.E80-A, No.11, pp.2201-2208, November 1997.

[19] Y. Komiya, T. Ohishi, and T. Matsumoto, "A Pen Input On-Line Signature Verifier Integrating Position, Pressure and Inclination Trajectories," IEICE Trans. Inf. and Syst., Vol.E84-D, No.7, pp.833-838, July 2001.

[20] R. Osada, S. Ozaki, T. Aoki and H. Yasuda, "A Real Time Personal Verification System Based on Individual Characteristic Extraction of Hand Motion," Trans. IEICE, Vol.J84-D-II, No.2, pp.258-265, February 2001 (in Japanese).

[21] A. K. Jain, L. Hong, and S. Pankanti, "Biometrics Identification," Communications of the ACM, Vol.43, No.2, pp.91-98, February 2000.

[22] B. Schneier, "Inside Risks: The Uses and Abuses of Biometrics," Communications of the ACM, Vol.42, No.8, pp.136-, August 1999.

[23] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing Security and Privacy in Biometrics-based Authentication Systems", IBM Systems Journal, Vol.40, No.3, pp.614-634, 2001.

[24] L. Hong, A. K. Jain, and S. Pankanti, "Can Multibiometrics Improve Performance?," Proc. AutoID '99, pp.59-64, October 1999.

[25] Y. Baba, H. Nakamura, T. Fujii, T. Sadakane, and N. Okazaki, "Evaluation of Remote Authentication System Using Fingerprint," Proc. APSITT '99, Ulaanbaatar, August 1999.

[26] "Secure Socket Layer," Netscape Communications.

[27] C. Rigney, S. Willens, A. Rubens, and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)," RFC 2865, IETF, June 2000.

Integrated person tracking using stereo, color, and pattern detection.

T. Darrell, G. Gordon, M. Harville, J. Woodfill
Interval Research Corp.
1801C Page Mill Road
Palo Alto CA 94304
trevor,gaile,harville,woodfill@interval.com

Abstract

We present an approach to real-time person tracking in crowded and/or unknown environments using multi-modal integration. We combine stereo, color, and face detection modules into a single robust system, and show an initial application in an interactive, face-responsive display. Dense, real-time stereo processing is used to isolate users from other objects and people in the background. Skin-hue classification identifies and tracks likely body parts within the silhouette of a user. Face pattern detection discriminates and localizes the face within the identified body parts. Faces and bodies of users are tracked over several temporal scales: short-term (user stays within the field of view), medium-term (user exits/reenters within minutes), and long term (user returns after hours or days). Short-term tracking is performed using simple region position and size correspondences, while medium and long-term tracking are based on statistics of user appearance. We discuss the failure modes of each individual module, describe our integration method, and report results with the complete system in trials with thousands of users.

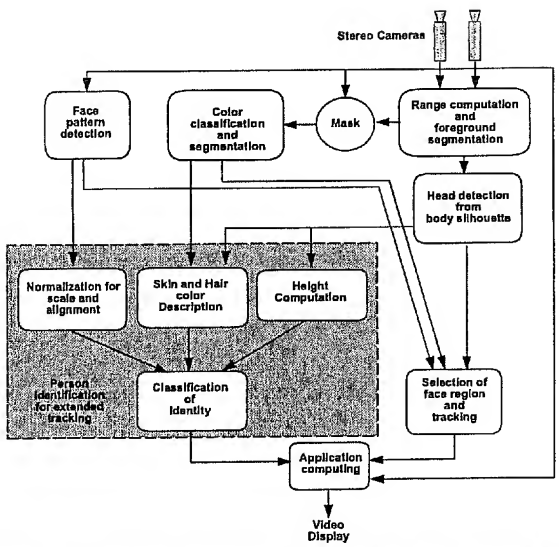

Figure 1. System overview showing the relationship of each modality with detection and short-term tracking, and with long-term tracking/identification.

1. Introduction

The creation of displays or environments which passively observe and react to people is an exciting challenge for computer vision [4, 6]. Faces and bodies are central to human communication and yet machines have been largely blind to their presence in real-time, unconstrained environments.

Often, computer vision systems for person tracking exploit a single visual processing technique to locate and track user features. These systems can be non-robust to real-world conditions with multiple people and/or moving backgrounds. Additionally, tracking is usually performed only over a single, short time scale: a person model is typically based only on an unbroken sequence of user observations, and is reset when the user is occluded or leaves the scene temporarily.

We have created a visual person tracking system which achieves robust performance through the integration of multiple visual processing modalities and by tracking over multiple temporal scales. With each modality alone it is possible to track a user under optimal conditions, but each also has, in our experience, substantial failure modes in unconstrained environments. Fortunately these failure modes are often independent, and by combining modules in simple ways we can build a system with overall robust performance.

In the following sections we describe our tracking framework and the three vision processing modalities used. We then describe an initial application of our system: a face-responsive, interactive video display. Finally we show the

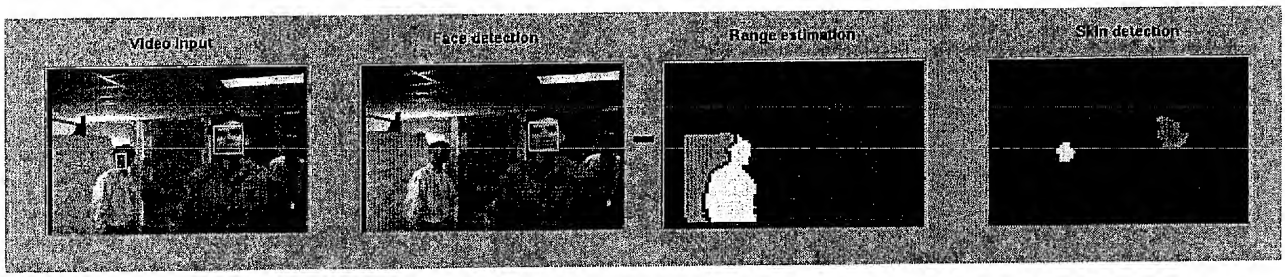

Figure 2. Output of vision processing modules: input image, face pattern detection output, connected components recovered from stereo range data, and flesh hue regions from skin hue classification. Boxes have been drawn on the faces of the two tracked users in the input image; the rightmost person in the image is beyond the workspace of the system.

results of our system when deployed with naive users, and analyze both the qualitative success of the application and the quantitative performance of our tracking algorithms.

2. Tracking framework

A person tracking system for interactive environments has several desired criteria: it should operate in real-time, be robust to multiple users and changing background, provide a relatively rich visual description of the users, and be able to track people when they are occluded or momentarily leave the scene. We achieve these goals through the use of multi-modal integration and multi-scale temporal tracking.

We base our system on three primary visual processing modules: depth estimation, color segmentation, and intensity pattern classification (see Figure 1). As described in more detail below, depth information is estimated using a dense real-time stereo technique and allows easy segmentation of the user from other people and background objects. An intensity-invariant color classifier detects regions of flesh tone on the user and is used to identify likely body part regions such as face and hands. A face detection module is used to discriminate head regions from hands and other tracked body parts.

Figure 2 shows the output of the three vision processing modules. As a person tracker, each is individually fragile: notebooks are indistinguishable from faces in range silhouette, flesh color signs or clothes fool color-only trackers, and face pattern detectors typically are slower and only work with relatively canonical poses and expressions. However, when integrated together these modules can yield robust, fast tracking performance.

Tracking is performed in our system on three different time-scales: short-range (frame to frame while the person is visible), medium-range (when the person is momentarily occluded or leaves the field of view for a few minutes), and long range (when the person is absent for hours, days or more.) Long-term tracking can be thought of as a person identification task, where the database is formed from the set of previous users. For short-term tracking we simply compute region correspondences specific to each processing modality based on region position and size. Multi-modal integration is performed using the history of short-term tracked regions from each modality, yielding a representation of the user's body shape and face location.

For medium and long-range tracking, we rely on a statistical model of multi-modal appearance to resolve correspondences between tracked users. In addition to body shape and face location, and color of hair, skin, and clothes is recorded at each time step. We record the average value and covariance of represented features, and use them for matching. For medium-term tracking, lighting constancy and stable clothing color are assumed; for long-term tracking we adjust for changing lighting and do not include clothing in the match criteria.

In the next section, we discuss module specific processing, including classification, segmentation/grouping, and short-term tracking. Following that, we present our integration scheme, and correspondence method for medium and long-term tracking.

3. Mode-specific processing

Pixel-wise classification, grouping and short-term tracking are performed independently in each modality. Stereo processing outputs a user's silhouette defined by range regions, color processing yields a set of skin color regions within range silhouette boundaries, and face processing returns a list of detected frontal face patterns; we describe each module in turn. Each mode also provides an independent estimate of head location and performs short-term tracking.

3.1. User silhouette from dense stereo

To compute a set of user silhouettes, we rely on a dense real-time stereo system. Video from a pair of cameras is used to estimate dense range using a technique based on the census transform [8]; we have implemented the census algorithm on a single PCI card, multi-FPGA reconfigurable computing engine [9]. This stereo system is capable of computing 24 stereo disparities on 320 by 240 images at

42 frames per second, or approximately 77 million pixel-disparities per second. These processing speeds compare favorably with other real-time stereo implementations such as [3].

Our segmentation and grouping technique proceeds in several stages of processing, as illustrated in Figure 3. We first smooth the raw range signal to reduce the effect of low confidence stereo disparities using a morphological closing operator. We then compute the response of a gradient operator on the smoothed range data and threshold at a critical value based on the observed noise level in our disparity data. Connected components analysis is applied to these regions of smoothly varying range. We return all connected components whose area exceeds a minimum threshold.

The range processing module provides these user silhouettes, as well as estimates of head location. A candidate head is placed below the maxima of the range profile. Head position is refined in the integration stage, as described below.

Disparity estimation, segmentation, and grouping are repeated independently at each time step; range silhouettes are tracked from frame to frame based on position and size constancy. The centroid and size of each new range silhouette is compared to silhouettes from the previous time step. "Short-term" correspondences are indicated using a greedy algorithm starting with the closest unmatched region; for each new region the closest old region within a minimum threshold is marked as the correspondence matches.

3.2. Skin color localization

Skin color is a useful cue for tracking people's faces and other body parts. We detect skin using a classification strategy which matches skin hue but is largely invariant to intensity or saturation, as this is robust to shading due to illumination and/or the absolute amount of skin pigment in a particular person.

We apply color segmentation processing to images obtained from one camera. Each image is initially represented with pixels corresponding to the red, green, and blue channels of the image, and is converted into a "log color-opponent" space. This space can directly represent the approximate hue of skin color, as well as its log intensity value. We convert (R, G, B) tuples into tuples of the form $(log(G), log(R) - log(G), log(B) - (log(R) + log(G))/2)$. Skin color is detected using a classifier with an empirically estimated Gaussian probability model of "skin" and "not-skin" in the log color-opponent color space. When a new pixel p is presented for classification, the likelihood ratio $P(p = skin)/P(p = non-skin)$ is computed as a classification score. Our color representation is similar to that used in [2], but we estimate our classification criteria from examples rather than apply hand-tuned parameters. For computational efficiency at run-time, we precompute a lookup table over all possible color values.

After the lookup table has been applied, segmentation and grouping analysis are performed on the classification score image. Similar to the range case, we use morphological smoothing, threshold above a critical value, and apply connected component computation. However, there is one difference: before smoothing we apply the low-gradient mask from the *range* modality. This restricts color regions to be grown only within the boundary of range regions; if spurious background skin hue is present in the background it will not adversely affect the shape of foreground skin color regions.

As with range processing, classification, segmentation, and grouping are repeated at each time step. Short-term tracking is performed on recovered color regions based on similar centroid position and region size. When a a color region changes size dramatically, we check to see if two regions merged, or if one region split into two. If so we record the identity of the split or merged regions, to be used in the integration stage as described below.

Skin color regions that are above the midline of their associated range component, and which are appropriately sized at the given depth to be heads, are labeled as candidate heads and passed to the integration phase.

3.3. Face pattern detection

To distinguish head from hands and other body parts, and to localize the face within a region containing the head, we use pattern recognition methods which directly model the statistical appearance of faces based on intensity.

We based our implementation of this module on the CMU face detector [7] library. This library implements a neural network which models the appearance of frontal faces in a scene, and is similar to the pattern recognition approach described in [5]. Both methods are trained on a structured set of examples of faces and non-faces.

Face detection is initially applied over the entire image; when one or more detections are recorded, they are passed directly as candidate head locations to the integration phase. Short term tracking is implemented by focusing search in a new frame within windows around the detected locations in the previous frame. If a new detection is found within such a window it is considered to be in short-term correspondence with the previous detection; if no new detection is found and the detection in the previous frame overlapped a color or range region, then the face detection is updated to move with that region (as long as it persists).

4. Integrated Tracking

Our integration method is designed to take advantage of each module's strengths: range is typically fast but coarse, color is fast and prone to false positives, and face pattern detection is slow and requires canonical pose and expression. We place priority on face detection hits, when available, and use color or range to update position from frame to frame.

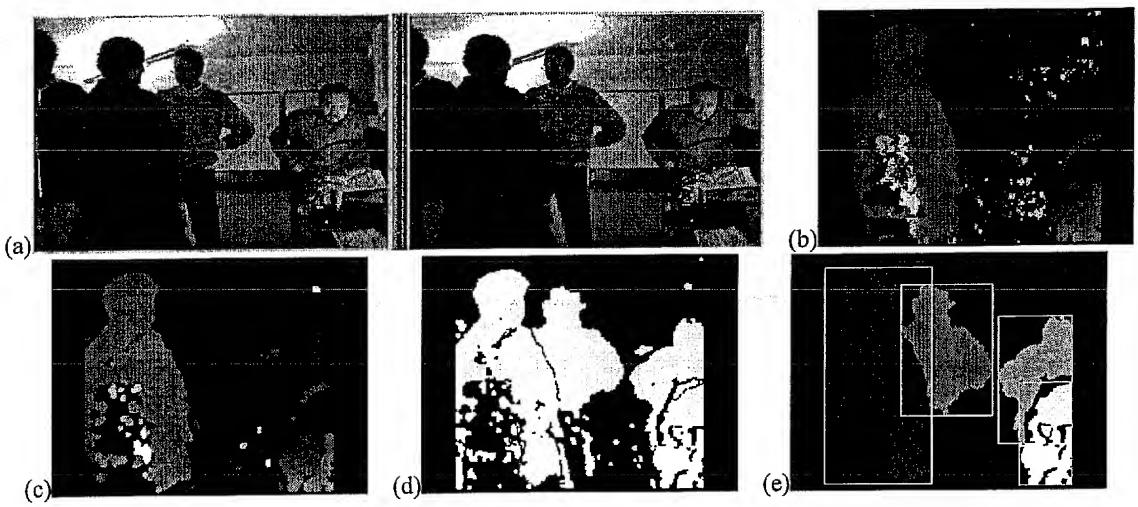

Figure 3. Stereo range processing to extract user silhouettes. (a) left/right image pair. (b) raw disparity computed using Census algorithm. (c) disparity after morphological smoothing. (d) regions of slowly varying disparity. (d) silhouettes recovered after connected components grouping.

For each range silhouette, we collect the range, color, and face detection candidate head features. As described above, when a candidate pattern detection head overlaps with a range or color candidate head, it persists and follows the range or color region. We record the relative offset of the face detection head with respect to the range or color head, and maintain that relationship in subsequent frames. This has the desired effect of allowing face detection to discriminate between head and hand regions in subsequent frames even when there may not be another face detection for several frames.

For each frame, we compute the location of a user's head on the range silhouette as follows: if a face detection candidate head is present, we return it; otherwise we return any location with overlapping range and color candidates, the location of the range candidate, or the location of a color candidate, in order of preference.

There is one special case in propagating face detection candidate heads. If the two color regions split or merge as described above, we take steps to allow the virtual face detection candidate head to follow the appropriate color region. We assume that the face is stationary between frames when deciding what color region to follow. If two regions have merged, the virtual detection follows the merged region, with offset such that the face's absolute position on the screen is the same as the previous frame. If two regions have split, the face follows the region closest to it's position in the previous frame. These heuristics are simple, but work in many cases where users are intermittently touching their face with their hands.

When the head location has been found, we update the estimate of head size. We have found that color is a relatively unreliable estimator of size; instead, we recompute size based on the results of the face detector and the range modules. When a face detection result has been found, we use it to determine the real size of the face. If no face detection hit has been found, we use an average model of real face size.

Our system can be configured in two modes: single- or multiple-person tracking. Singe-person mode is most appropriate for interactive games or kiosks which are restricted to a single user; multiple-person is more appropriate for general surveillance and monitoring applications. In single person mode, we return only a single range silhouette; we initially choose the closest range region, and then follow that region until it is no longer tracked in the short-term.

5. Long-term tracking

When users are momentarily occluded or exit the scene, short-term tracking will fail since position and size correspondences in the individual modules are unavailable. To track users over medium and long-term time scales, we rely on statistical appearance models. Each visual processing module computes an estimate of certain user attributes, which are expected to be stable over longer time periods. These attributes are averaged as long as the underlying range silhouette continues to be tracked in the short-term, and used in a classification stage to establish medium and long-term correspondences.

Like multi-modal person detection and tracking, multi-modal person appearance classification is more robust than classification systems based on a single data modality. Height, color, and face pattern each offer independent classification data and are accompanied by similarly indepen-

dent failure modes. Although face patterns are perhaps the most common data source for current passive person classification methods, it is unusual to incorporate height or color information in identification systems because they do not provide sufficient discrimination to justify their use alone. However, combined with each other and with face patterns, height and color can provide important cues to disambiguate otherwise similar people, or help classify people when only degraded data is available in other modes.

5.1. Observed attributes

In the range module, we estimate the height of the user and use this as an attribute of identity. Height is obtained by computing the median value of the highest point of the a user silhouette in 3-D. In the color module, we compute the average color of the skin and hair regions; we plan to also add a histogram of clothing color. We define the hair region to be those pixels above the face but on the range silhouette; clothing can be defined as all other silhouette pixels not labeled as skin or hair.

In the face detector, we record an image of the actual face pattern wherever the detector records a hit. When a region is identified as a face based on the face pattern detection algorithm, the face pattern (greyscale subimage) in the target region is normalized and then passed to the classification stage. For optimal classification, we want the scale, alignment, and view of detected faces to be comparable. We resize the pattern to normalize for size, and discard images which are not in canonical pose or expression, which is determined by normalized correlation with an average canonical face.

For "medium-term" tracking, e.g., over seconds or minutes of occlusion or absence, we rely on all of the above attributes. For "long-term" tracking, over hours or longer, we cannot rely on attributes which are not invariant with time of day or from day to day: we correct all color values with a mean color shift to account for changing illumination, and would exclude clothing color from the match computation.

5.2. Classification

In general, we compute statistics of these attributes while users are being tracked over the short-term, and compare against stored statistics of all previous tracked users.

When we observe a new person, we see if there is a previously tracked individual which could have generated the current observations. We find the previous individual most likely to have generated the new observations; if this probability is above a minimum threshold, we label the currently tracked region as being in correspondence with the previous individual. We integrate likelihood over time and modality: at time t, we find the identity estimate

$$u^* = \arg\max_j P(U_j|\omega) \quad (1)$$

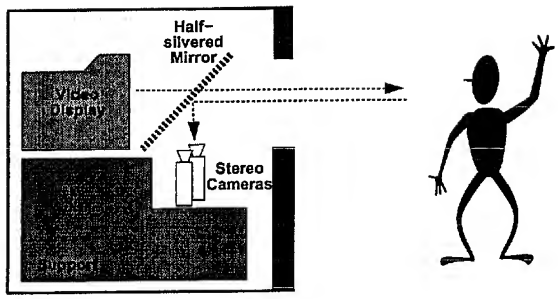

Figure 4. Display and viewing geometry: cameras and video-display share optical axis through a half-silvered mirror.

where

$$P(U_j|\omega) = P(U_j|F_0,...,F_t,H_0,...,H_t,C_0,...C_t) \quad (2)$$

where F_i,H_i, and C_i are the face pattern, height, and color observations at time i, and U_j are the saved statistics for person j. We restart time at $t = 0$ when a new range silhouette is tracked. For the purposes of this discussion we assume $P(U_j)$ is uniform across users. With Bayes rule and the assumption of modality independence, we have:

$$u^* = \arg\max_j(P(F_0,...,F_t|U_j) \\ P(H_0,...,H_t|U_j)P(C_0,...,C_t|U_j)) \quad (3)$$

Since our observations are independent of the observed noise in our sensor and segmentation routines, the posterior probabilities at different times may be considered independent. With this we can incrementally compute probability in each modality:

$$P(F_0,...,F_t|U_j) = P(F_0,...,F_{t-1}|U_j)P(F_t|U_j) \quad (4)$$

and similarly for range and color data.

We collect mean and covariance data for the observed user color data, and mean and variance of user height; the likelihoods $P(F_i|U_j)$ and $P(C_i|U_j)$ are computed assuming a Gaussian density model. For face pattern data, we store the size- and position-normalized mean pattern for each user, and approximate $P(F_t|C_p)$ with an empirically determined density which is a function of the normalized correlation of F_t with the the mean pattern for person j.

6. A Real-time Virtual Mirror Display

Our initial application of our integrated, multi-modal visual person tracking framework is to create a face-responsive visual display. We construct a video display where cameras observe the user from the same optical axis as used by the display, and send estimates of the 3-D head position of observers of the screen to the application program. One application we have explored using this display

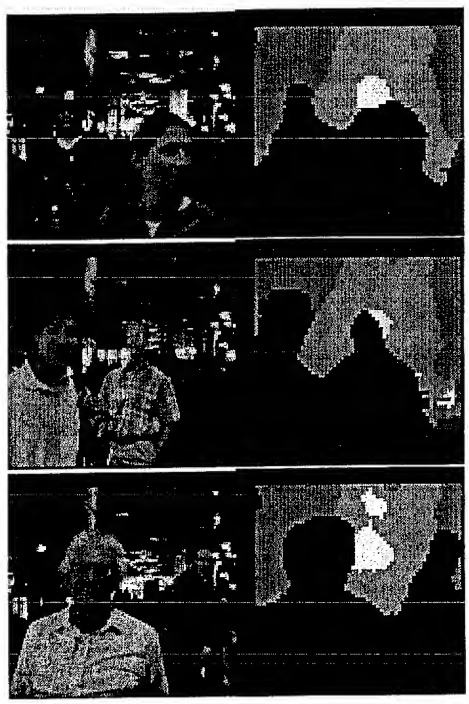

Figure 5. Color/Range stills of virtual mirror users collected during the SIGGRAPH '97 demonstration.

is an interactive graphics experience in which users' faces are distorted in real-time. The effect is a virtual fun-house mirror, but in which only the face regions are distorted.

We create a virtual mirror by placing cameras so that they share the same optical axis as a video display, using a half-silvered mirror to merge the two optical paths. The cameras view the user through a 45-degree half mirror, so that the user can view a video monitor while also looking straight into (but not seeing) the cameras. Video from one camera is displayed on the monitor after the application of various computer graphics distortion effects, so as to create a virtual mirror effect. Figure 4 shows the display and viewing geometry of our apparatus. Using video texture mapping and the OpenGL graphics system, we have implemented graphics methods to distort faces on the screen using one of the following special effects: spherical expansion, spherical shrinking, swirl, lateral expansion, and a vertical melting effect. This creates a novel and entertaining interactive visual experience where users get immediate visual feedback from their tracked faces.

Our system is currently implemented using three computer systems (one PC, two SGI O2), a large NTSC video monitor, stereo video cameras, a dedicated stereo computation PC board, and the half-mirror imaging apparatus. The full tracking system, including all vision and graphics processing, runs at approximately 12Hz.

Figure 6. Example distortion output from virtual mirror application.

7. Results

We first demonstrated our system at the SIGGRAPH Conference from August 3-8, 1997 [1]. An estimated 5000 people over 6 days used our system (approximately two new users per minute, over 42 hours of operation). The goal of the system in this application was to identify the 3-D position and size of a single user's head in the scene, and apply a distortion effect in real-time only over the region of the image containing the user's face. The distorted image was then displayed on the virtual mirror screen. The system tracked the user while he or she was in the frame, and then switched to a new user.

Qualitatively, the system was a complete success. Our tracking results were able to localize video distortion effects on the user's face, and overall the system was interesting and fun for people to use. Figure 6 shows a typical final image displayed on the virtual mirror. The system performed well with both single users and crowded conditions; the background environment was quite visually noisy, with many spurious lighting effects being randomly projected throughout the conference hall, including onto the people being tracked by our system.

7.1. Evaluation

We quantitatively evaluated the performance of our system using three off-line datasets: a set of stills captured at SIGGRAPH to evaluate detection performance, a set of stills of users in our laboratory, and a set of appearance statistics gathered from users in our laboratory who interacted with the system over several days. (Unfortunately we were not able to obtain observations of the same users across multiple days at the SIGGRAPH demonstration.)

We collected stills of users interacting with our system every 15 seconds over a period of 3 hours at the SIGGRAPH demonstration. At each sample point we captured both a color image of the scene and a greyscale image of the output of the range module after disparity smoothing. We discarded images with no users present, yielding approx-

Modules Enabled			SIGGRAPH data	Lab data	Overall
Color	*Range*	*Pattern*			
√	√	√	97%	96%	97%
	√	√	97%	95%	96%
√	√		97%	93%	95%
	√		97%	90%	94%
√		√	92%	93%	92%
√			90%	89%	90%
		√	22% †	80%	44%

Table 1. Face detection and localization results on SIGGRAPH and Lab datasets using different combinations of input modules, ordered by increasing error rate. (†) This data included many images which were smaller than the size range the pattern module was trained to detect.

imately 300 registered color/range pairs. Figure 5 shows examples of the collected stills. We also collected a similar set of approximately 200 registered range/color stills of users of the system while on display in our laboratory, similar to the images in Figures 2 and 3(a). Table 1 summarizes the single-person detection results we obtained on these test images. A correct match was defined when the corners of the estimated face region were sufficiently close to manually entered ground truth (within $\frac{1}{4}$ of the face size). Overall, when all modules were functioning, we achieved a success rate of 97%; when the color and/or face detection module was removed, performance was still above 93%, indicating the power of the range cue for detecting likely head locations.

To evaluate our longer term tracking performance we used statistics gathered from 25 people in our laboratory who visited our display several times on different days. People's hairstyle, clothing, and the exterior illumination conditions varied between the times data were collected. We tested whether our system was able to correctly identify users when they returned to the display. In general, our results were better for medium term tracking (intra-day) than for long term (inter-day) tracking, as would be expected. Table 2 shows the extended tracking results: the correct classification percentage is shown for each modality and for the combined observations from all modes. This table reflects the recognition rate using all of the data from each short-term tracking session: on average, users were tracked for 15 seconds before short-term tracking failed or they exited the workspace.

By integrating modes we were able to correctly establish correspondences between tracked users in all of the medium-term cases, which typically involved temporal gaps between 10 and 100 seconds. In the long-term cases, which typically reflected gaps of one day, integrated performance was 87%. A more complete description of medium- and

Performance	Medium-term (intra-day)	Long-term (inter-day)
Height	44%	20%
Color	84%	63%
Face pattern	84%	67%
Multi-modal	100%	87%

Table 2. Extended tracking performance: correct identification rate at end of session.

long-term performance is shown in Figure 7(a) and Figure 7(d), respectively. These figures show the recognition rate vs rank threshold, i.e., the percentage of time the correct person was above a given rank in the ordered likelihood list of predicted users. We also measured our performance over time: Figures 7(b) and 7(e) compare the performance versus rank threshold at 4 different times during each testing session. Here we show only the multi-modal results; as expected, identification becomes more reliable over time as more data is collected. Figures 7(c) and 7(f) show the rank of the correct person over time, averaged across all test sessions; correct identification (average rank equals one) is almost always achieved within one second in the medium-term case, and within three seconds in the long-term case.

7.2. Discussion

We draw two main conclusions from the detection results; first, that range data is a powerful cue to detecting heads in complex scenes. Second, integration is useful: in almost every case, the addition of modules improved system performance. Performance was generally high, but individual module results varied considerably across datasets. In particular the face pattern module fared relatively poorly on the SIGGRAPH dataset. We believe that this is largely due to the small size and poor illumination of many of the faces in these images. Additionally, in both datasets our application encouraged people to make exaggerated expressions, which was beyond the scope of the training for this module.

In contrast, for extended tracking it is clear from these results that the face pattern is the most valuable of the three modes when we consider all the data available during the session. Face pattern data is most discriminating at the *end* of the test session; however, the other modalities are dominant early in the session. The face detection module operates more slowly than the other modes, so the face pattern data is not available immediately and accumulates at a slower rate. Therefore, in the first few seconds the overall performance of the extended tracking system is due primarily to color and height data, and far exceeds the performance based on face pattern alone.

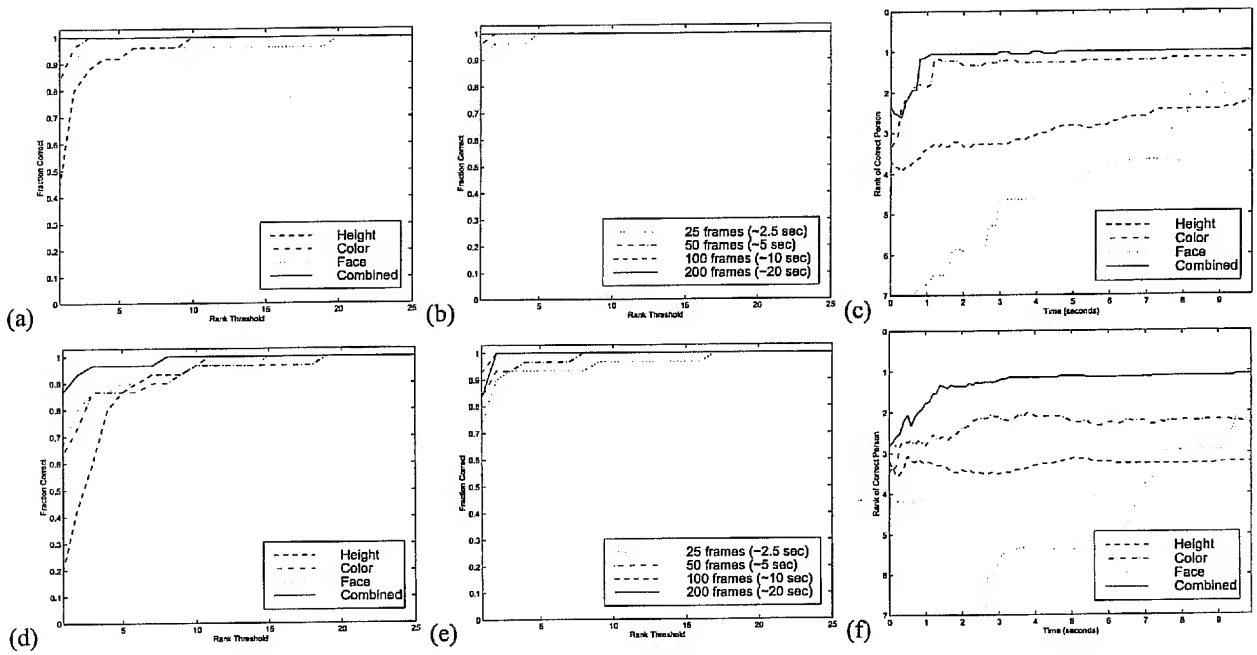

Figure 7. Medium-term tracking: (a) performance vs rank threshold, results for each modality separately and then in combination, (b) multi-modal performance vs rank threshold at 4 different time samples during a session, (c) average rank of correct person over time. (d,e,f) Results for long-term tracking.

8. Conclusion

We have demonstrated a system which can respond to a user's face in real-time using completely passive and non-invasive techniques. Robust performance is achieved through the integration of three key modules: depth estimation to eliminate background effects, color classification for fast tracking, and pattern detection to discriminate the face from other body parts. We use descriptions of the user computed from the same modalities to track over longer time scales when the user is occluded or leaves the scene. Our system has application in interactive entertainment, telepresence/virtual environments, and intelligent kiosks which respond selectively according to the presence, pose, and identity of a user. We hope these and related techniques can eventually balance the I/O bandwidth between typical users and computer systems, so that they can control complicated virtual graphics objects and agents directly with their own expression.

References

[1] Darrell, T., Gordon, G., Woodfill, W., Baker, H., A Magic Morphin Mirror, SIGGRAPH '97 Visual Proceedings, ACM Press. 1997.

[2] Margaret Fleck, David Forsyth, and Chris Bregler (1996) "Finding Naked People," European Conference on Computer Vision , Volume II, pp. 592-602. 1996.

[3] Kanade, T., Yoshida, A., Oda, K., Kano, H., and Tanaka, M., "A Video-Rate Stereo Machine and Its New Applications", Computer Vision and Pattern Recognition Conference, San Francisco, CA, 1996.

[4] Maes, P., Darrell, T., Blumberg, B., and Pentland, A.P., "The ALIVE System: Wireless, Full-Body, Interaction with Autonomous Agents". ACM Multimedia Systems: Special Issue on on Multimedia and Multisensory Virtual Worlds, Sprint 1996.

[5] Poggio, T., Sung, K.K., Example-based learning for view-based human face detection. Proceedings of the ARPA IU Workshop '94, II:843-850. 1994.

[6] Rehg, J., Loughlin, M., and Waters, K., "Vision for a Smart Kiosk", Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR-97, pp. 690-696. IEEE Computer Society Press. 1997.

[7] Rowley, H., Baluja, S., and Kanade, T., Neural Network-Based Face Detection, Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR-96, pp. 203-207,. IEEE Computer Society Press. 1996.

[8] Zabih, R., and Woodfill, J., Non-parametric Local Transforms for Computing Visual Correspondence, Proceedings of the third European Conference on Computer Vision, Stockholm, pp. 151 - 158. May 1994.

[9] Woodfill, J., and Von Herzen, B., Real-Time Stereo Vision on the PARTS Reconfigurable Computer, Proceedings IEEE Symposium on Field-Programmable Custom Computing Machines, Napa, pp. 242-250, April 1997.

UNITED STATES PATENT AND TRADEMARK OFFICE

Sony-29200

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

COPY

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/763,654	01/22/2004	Edward Eytchison	Sony-05400	9392

28960 7590 10/16/2008

HAVERSTOCK & OWENS LLP
162 N WOLFE ROAD
SUNNYVALE, CA 94086

RECD OCT 20 2008

EXAMINER	
JUNG, DAVID YIUK	
ART UNIT	PAPER NUMBER
2434	

MAIL DATE	DELIVERY MODE
10/16/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/763,654	EYTCHISON ET AL.	
	Examiner	Art Unit	
	David Y. Jung	2434	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on 6/26/2008.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) 1-34 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) 1-34 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

CLAIMS PRESENTED

Claims 1-31 are presented.

Response to Arguments

Applicant's arguments filed have been fully considered but they are not persuasive.

At page 9, Applicant asserts that: "In contrast to the teachings of Darrell, Davis and their combination, the user identification system of the present invention stores one or more historic, idiosyncratic activity patterns as user action identification profiles. The method and apparatus of the present invention monitors the current user's electronic device inputs to determine the current user's idiosyncratic activity pattern. If the detected activity pattern is deemed sufficiently close to a historic, idiosyncratic activity pattern associated with a particular user, then the current user is identified as the historic, particular user. Monitored activity patterns include one or a combination of the user's selected content, the user's manner of selecting the content, the context in which the user makes certain inputs. In some instances, the module monitors the physical manner in which the user operates the electronic device. For example, two users will make keypad inputs at different speeds. In one embodiment, once the current user's identification is established, either passively by a comparator module or actively by requesting a specific user input, a password, the identification system updates the identified user's action identification profile." See Page 9 of the Remarks Section of the

Applicant's filing. Yet, there is no claim that recites all of these features. Not one claim. Despite this, Applicant argues that the references fail to teach all of these features. Thus, in response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (e.g., the typing at different speeds) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

As for Applicant's argument (e.g., pages 10-11) regarding the "one activity performed by the particular unique user" not being taught by the prior art, Applicant has cited sections of Davis and sections of Darrell in which the particular sections, when read in isolation from other sections, do not teach the particular recitation. See pages 10-11 of Applicant's Remarks. This is, of course, logically against the very idea of an obviousness rejection; the rejection was concerned about the sections and reasonable readings of Davis and Darrell in which the claimed invention was taught. The actual rejection was not accurately addressed. The actual rejection did not deal with how to combine the portions of Davis and Darrell so as to not teach the claimed invention. While Davis and Darrell do lend themselves to many different reasonable combinations of their teachings, the actual rejection dealt with how to combine the teachings of Davis and Darrell so as to teach the claimed invention. This much is clear. Davis was cited for identifying the activity. Darrell was cited for identifying the particular unique user.

The combination of the teachings teach the "one activity performed by the particular unique user."

Applicant may have interpreted the rejection to have dealt with a physical combination of Davis and Darrell (as would be reasonable for mechanical objects). The rejection did not deal with any direct combination of the devices. Instead, the teachings of Davis and the teachings of Darrell were to be combined. In any case, any record of misunderstandings and mistaken readings of file history (e.g., regarding rejections in Office Actions) should be pointed out and corrected.

Therefore, Applicant is respectfully requested to submit further arguments or amendments or other appropriate responses.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Regarding all presented claims, the relied references are as noted in the previous Office Action. Please see the previous Office Action for detailed citation.

Claims 1-21, 25-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Darrell and Davis.

Regarding claim 1, Davis teaches "A method of identifying a user comprising: detecting a user's electronic device activity pattern; comparing the detected activity pattern against a plurality of user action identification profiles stored in a memory_ device, wherein each user action identification profile is associated with a [ ] unique [ ] by at least one activity performed by the particular unique user; and using the comparing [ ] (section 3 Application Space, e.g., credit card fraud detection which identifies patterns of credit card use)."

These passages of Davis do not teach "particular ... user" and "to identify the current user as being one of the particular users" in the sense of the claim.

Darrell teaches "particular ... user" and "to identify the current user as being one of the particular users (Figure 1, i.e., tracking and identification)" for the motivation of better tracking persons (section 1 Introduction, i.e., locate and track user).

Hence, it would have been obvious to those of ordinary skill in the art at the time of the claimed invention to combine the teachings of Darrell and Davis for the motivation noted in the previous paragraphs so as to teach the claimed invention.

Claims 1, 10, 11, 21, 27 are independent claims.

The special features of claim 10 (detection etc.,), claim 11 (system etc.,), claim 21 (storing the activity pattern etc.,), claim 27 (identification system with modules etc.,) are taught by Darrell (section 4 Integrated Tracking, especially the second paragraph concerning collecting data and detection).

Regarding claims 2, 5-9, 12, 15-19, 25, 28, 29, 31 (various user identifying), these features are taught by Darrell (section 4 Integrated Tracking, especially the second paragraph concerning collecting data and detection)

Regarding claims 3, 4, 13, 14, 20, 26, 30 (various scoring as to whether matches are reasonable), these features are taught by Davis (section 3. Application Space, e.g., credit card fraud detection which identifies patterns of credit card use)."

Claims 22-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Darrell and Davis and Seno.

Claims 22-24 depend from claim 21. Darrell and Davis teach as noted in the rejection of claim 21.

Thus, Darrell and Davis teach all but the special features of claims 22-24 (biometric features and password, etc.).

The special features of claims 22-24 (biometric features and password, etc.) are taught by Seno (section I. Introduction, i.e., finger print, iris, passwords, etc.) for the motivation of user authentication ((section I. Introduction).

Hence, it would have been obvious to those of ordinary skill in the art at the time of the claimed invention to combine the teachings of Darrell and Davis and Seno for the motivation noted in the previous paragraphs so as to teach the claimed invention.

Conclusion

The art made of record and not relied upon is considered pertinent to applicant's disclosure. The art disclosed general background.

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Points of Contact

Any response to this action should be mailed to:

Commissioner for Patents

Alexandria, VA 22313

or faxed to:

(571) 273-8300, (for formal communications intended for entry)

Or:

(571) 273-3836 (for informal or draft communications, please label "PROPOSED" or "DRAFT")

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David Jung whose telephone number is (571) 272-3836 or Kambiz Zand whose telephone number is (571) 272-3811.

/David Y Jung/

Acting Examiner of Art Unit 2134

David Jung

David Jung

Patent Examiner

10/15/08